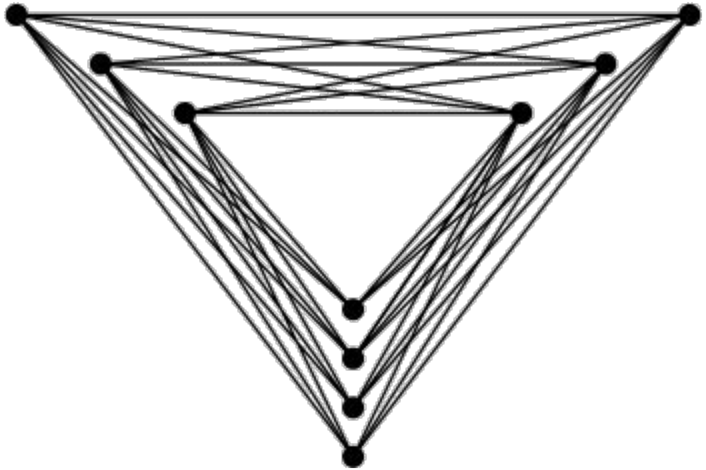
Departamento de Ingeniería Informática
Universidad de Santiago de Chile

Teoría de la Computación

Primer semestre 2024



Daniel Vega Araya

Santiago, 2024

Motivación

Dos personas conversan en el metro, y ud escucha parte de su conversación...

Motivación

Dos personas conversan en el metro, y ud escucha parte de su conversación...

Persona 1: con todos los eventos que han ocurrido, el dólar ha sufrido una fuerte alza.

Motivación

Dos personas conversan en el metro, y ud escucha parte de su conversación...

Persona 1: con todos los eventos que han ocurrido, el dólar ha sufrido una fuerte alza.

Persona 2: claro, y además los economistas siempre dicen que si el dólar tiende al alza, se incrementa el costo de la tecnología...

Motivación

Dos personas conversan en el metro, y ud escucha parte de su conversación...

Persona 1: con todos los eventos que han ocurrido, el dólar ha sufrido una fuerte alza.

Persona 2: claro, y además los economistas siempre dicen que si el dólar tiende al alza, se incrementa el costo de la tecnología.... Así que, “claramente” no es el momento de comprar *chiches* tecnológicos, ¿no te parece?.

Motivación

con todos los eventos que han ocurrido, el dólar ha sufrido una fuerte alza

si el dólar tiende al alza, se incrementa el costo de la tecnología

“claramente” no es el momento de comprar *chiches* tecnológicos

Motivación

con todos los eventos que han ocurrido, el dólar ha sufrido una fuerte alza

si el dólar tiende al alza, se incrementa el costo de la tecnología

→ "claramente" no es el momento de comprar *chiches* tecnológicos

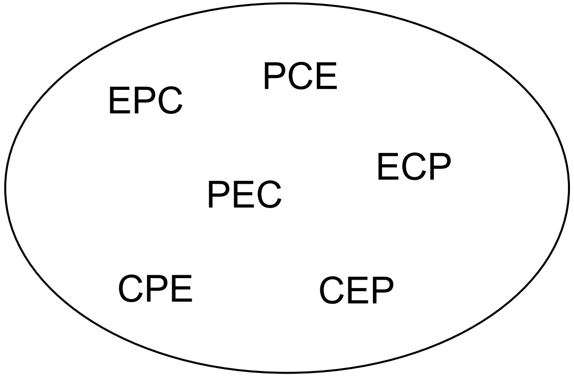
Motivación

En un restaurant una primera mesera registra la solicitud de una familia, el padre ordena pescado (P), la madre ensalada (E) y el hijo carne (C). Al llegar una segunda mesera con las órdenes, ¿qué sucede?

Motivación

En un restaurant una primera mesera registra la solicitud de una familia, el padre ordena pescado (P), la madre ensalada (E) y el hijo carne (C). Al llegar una segunda mesera con las órdenes, ¿qué sucede?

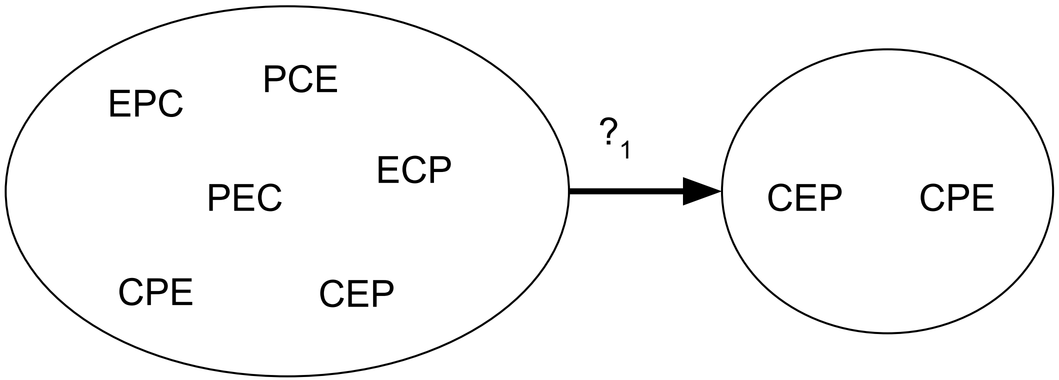
Escenario….



Motivación

En un restaurant una primera mesera registra la solicitud de una familia, el padre ordena pescado (P), la madre ensalada (E) y el hijo carne (C). Al llegar una segunda mesera con las órdenes, ¿qué sucede?

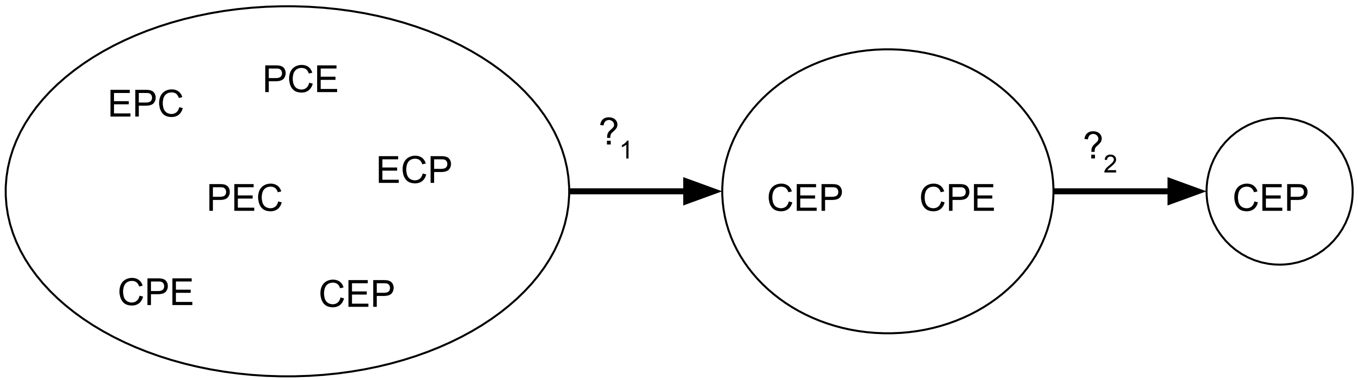
Escenario....



Motivación

En un restaurant una primera mesera registra la solicitud de una familia, el padre ordena pescado (P), la madre ensalada (E) y el hijo carne (C). Al llegar una segunda mesera con las órdenes, ¿qué sucede?

Escenario….



Motivación

- Observación (contexto)
- Comunicación
- Inferencia (conclusión a partir de premisas)

Introducción

Algunas formas de razonamiento y tipos de lógica

- Razonamiento matemático
- Razonamiento con sentido común
- Razonamiento inductivo

- Lógica Difusa
- Lógica Multivaluada
- Lógica Probabilística
- Lógica Modal

...

Introducción

¿qué es y cómo se caracteriza una lógica?

- Sintaxis del lenguaje
- Semántica
- Sistema deductivo
- Correspondencia entre sintaxis y semántica
- Implementación

Introducción

Metalenguaje: lenguaje en el que se desenvuelve la investigación (en el que se habla)

Lenguaje objeto: lenguaje investigado (del que se habla)

Ejemplo: El estudio de la gramática alemana en un país hispanohablante

Introducción

Inferencia y sentencias que conllevan expresiones ‘**y**’, ‘**o**’, ‘**no**’, ‘**si, entonces**’

Intuitivamente argumento y consecuencia

Si toma el medicamento, se recuperará
ud. no se está tomando el medicamento

Por lo tanto, ud. no se recuperará

conclusión a partir de sentencias lógicas o **proposiciones...**

Introducción

Proposición: enunciado declarativo (verdadero o falso), pero no ambos a la vez.

- Sentencias que son proposiciones

    a) Mi mascota es un hamster

    b) La suma de los números 3 y 5 es 8

- Sentencias que no son proposiciones

    a) ¿Me podrías pasar el libro?

    b) Lo voy a pensar

Introducción

Silogismo: Razonamiento que está formado por dos premisas y una conclusión que es el resultado lógico que se deduce de las dos premisas.

Ejemplo:

Premisa Universal: Las aves tienen plumas

Premisa Particular: Mi **pato** tiene plumas

Conclusión: Mi **pato** es un ave.

Introducción

Silogismo: Razonamiento que está formado por dos premisas y una conclusión que es el resultado lógico que se deduce de las dos premisas.

Ejemplo:

Premisa Universal: Las aves tienen plumas

Premisa Particular: Mi **pato** tiene plumas

Conclusión: Mi **pato** es un ave.

Premisa Universal: Las aves tienen plumas

Premisa Particular: Mi **almohada** tiene plumas

Conclusión: Mi **almohada** es un ave.

[INICIO - REPASO]

Simbolización de proposiciones

- **Término de enlace**

  o

  y

  no

  si…, entonces…

- **Proposiciones**

  - **Atómicas**

    Hoy es lunes

    Hay clases

  - **Moleculares**

    Hoy es lunes y hay clases

Simbolización de proposiciones

Para cada frase, indicar si es una proposición atómica o molecular e indicar según corresponda, el o los términos de enlace:

1. La comida será hoy a las 15:00 en punto.
2. La música es muy suave o la puerta está cerrada.
3. A este perro negro le gusta cazar conejos.
4. La luna no está hecha de queso.
5. Si x = 0, entonces y = 1.
6. Las personas mayores no están en la lista antes que los jóvenes.
7. Puede amanecer despejado en la playa.
8. x + y <= 5.

Simbolización de proposiciones

Paréntesis

(________) y (________)

(________) o (________)

Si (________), entonces (________)

No (________)

Simbolización de proposiciones

Paréntesis y sus variantes

(________) y (________)
A la vez (________) y (________)

(________) o (________)
O (________) o (________)

Si (________), entonces (________)
Si (________), (________)

No (________)
No ocurre que (________)

Simbolización de proposiciones

Proposiciones y variables proposicionales:

p = “la comida está congelada”
q = “tengo hambre”
r = “el microondas calienta rápido la comida”
s = “puedo comer enseguida”

Conectores lógicos:

~ (No),
$\vee$ (o),
$\wedge$ (y),
$\rightarrow$ (Si..., entonces...)

Ejemplos:

1. Tengo hambre y la comida está congelada.
2. Como el microondas calienta rápido la comida, puedo comer enseguida.
3. Tengo hambre pero puedo comer enseguida.

Simbolización de proposiciones

Proposiciones y variables proposicionales:

p = “la comida está congelada”
q = “tengo hambre”
r = “el microondas calienta rápido la comida”
s = “puedo comer enseguida”

Conectores lógicos:

~ (No),
$\vee$ (o),
$\wedge$ (y),
$\rightarrow$ (Si…, entonces…)

Ejemplos:

1. Tengo hambre y la comida está congelada.
   $(q \wedge p)$
2. Como el microondas calienta rápido la comida, puedo comer enseguida.

3. Tengo hambre pero puedo comer enseguida.

Simbolización de proposiciones

Proposiciones y variables proposicionales:

p = “la comida está congelada”
q = “tengo hambre”
r = “el microondas calienta rápido la comida”
s = “puedo comer enseguida”

Conectores lógicos:

~ (No),
$\lor$ (o),
$\land$ (y),
$\rightarrow$ (Si..., entonces...)

Ejemplos:

1. Tengo hambre y la comida está congelada.
   $(q \land p)$
2. Como el microondas calienta rápido la comida, puedo comer enseguida.
   $(r \rightarrow s)$
3. Tengo hambre pero puedo comer enseguida.

Simbolización de proposiciones

Proposiciones y variables proposicionales:

p = "la comida está congelada"
q = "tengo hambre"
r = "el microondas calienta rápido la comida"
s = "puedo comer enseguida"

Conectores lógicos:

~ (No),
$\vee$ (o),
$\wedge$ (y),
$\rightarrow$ (Si..., entonces...)

Ejemplos:

1. Tengo hambre y la comida está congelada.
   $(q \wedge p)$
2. Como el microondas calienta rápido la comida, puedo comer enseguida.
   $(r \rightarrow s)$
3. Tengo hambre pero puedo comer enseguida.
   $(q \wedge s)$

Simbolización de proposiciones

Ejercicios:

Sean:

w = “él está equivocado”
r = “yo tengo razón”
s = “quedaré sorprendido”

1. O él está equivocado y yo tengo razón, o quedaré sorprendido.

Simbolización de proposiciones

Ejercicios:

Sean:

w = “él está equivocado”
r = “yo tengo razón”
s = “quedaré sorprendido”

1. O él está equivocado y yo tengo razón, o quedaré sorprendido. $(w \land r) \lor s$

2. Él está equivocado, y o yo tengo razón o quedaré sorprendido.

Simbolización de proposiciones

Ejercicios:

Sean:

w = “él está equivocado”
r = “yo tengo razón”
s = “quedaré sorprendido”

1. O él está equivocado y yo tengo razón, o quedaré sorprendido. $(w \wedge r) \vee s$

2. Él está equivocado, y o yo tengo razón o quedaré sorprendido. $w \wedge (r \vee s)$

3. Él está equivocado y yo tengo razón o quedaré sorprendido.

Simbolización de proposiciones

Ejercicios:

Sean:

w = "él está equivocado"
r = "yo tengo razón"
s = "quedaré sorprendido"

1. O él está equivocado y yo tengo razón, o quedaré sorprendido. $(w \wedge r) \vee s$
2. Él está equivocado, y o yo tengo razón o quedaré sorprendido. $w \wedge (r \vee s)$
3. **Él está equivocado y yo tengo razón o quedaré sorprendido. (ambigua)**

Simbolización de proposiciones

Eliminación de paréntesis

- Regla 1: el $\rightarrow$ es más potente que los otros términos de enlace

$(p \land q) \rightarrow r$

$p \land q \rightarrow r$

$p \rightarrow (q \lor r)$

$p \rightarrow q \lor r$

pero,

$(p \rightarrow q) \lor r$

$p \rightarrow (q \rightarrow r)$

Simbolización de proposiciones

Eliminación de paréntesis

- Regla 2: el signo de negación es el más débil

  $(\sim q)$

  $\sim q$

  pero,

  $\sim(p \land q)$

Finalmente, $\land$ y $\lor$ son igualmente fuertes.

Simbolización de proposiciones

Ejercicios

Escriba empleando proposiciones:

1. Si x es menor que tres, entonces es menor que cuatro.
2. x es mayor que cinco.
3. No somos capaces de hacer ejercicio todos los días.

Traducir las siguientes proposiciones lógicas al lenguaje cotidiano

1. $(\sim q \land p) \lor s \rightarrow r$
2. $p \rightarrow \sim q \lor r$

[FIN - REPASO]

Introducción

Consideremos el siguiente argumento:

Si el tren llega tarde y no hay taxis en la estación, entonces José llega tarde a su reunión. José no llega tarde a su reunión. El tren llegó tarde. Por lo tanto, había taxis en la estación.

Si consideramos,

p = “El tren llega tarde”
q = “hay taxis en la estación”
r = “José llega tarde a su reunión”

¿qué podemos decir respecto a su validez?.

Introducción

Consideremos el siguiente argumento:

Si el tren llega tarde y no hay taxis en la estación, entonces José llega tarde a su reunión. José no llega tarde a su reunión. El tren llegó tarde. Por lo tanto, había taxis en la estación.

Si consideramos,

p = "El tren llega tarde"
q = "hay taxis en la estación"
r = "José llega tarde a su reunión"

¿qué podemos decir respecto a su validez?. Lo anterior nos motiva a preguntar ¿cómo construir un procedimiento tal que podamos razonar a través de proposiciones y con ello determinar su validez?

Deducción natural

En deducción natural, existe una colección de *reglas de prueba*, las que permiten inferir fórmulas a partir de otras fórmulas. Así, aplicando sucesivamente las reglas de prueba, podemos inferir una conclusión a partir de un conjunto de premisas.

Supongamos que tenemos un conjunto de fórmulas $\varphi_1,\varphi_2,\ldots,\varphi_n$, el que llamaremos premisas, y otra fórmula, ψ, la cual será llamada conclusión. Al aplicar reglas de prueba a las premisas, esperamos obtener algunas fórmulas más, las que al aplicar sobre ellas las reglas de prueba nos permitan eventualmente llegar a la conclusión. Lo anterior será denotado como:

$$\varphi_1,\varphi_2,\ldots,\varphi_n \vdash \psi$$

Esta expresión es llamada **secuente**; el cual es válido si es posible encontrar una prueba para él.

Deducción natural

Consideremos el siguiente argumento:

Si el tren llega tarde y no hay taxis en la estación, entonces José llega tarde a su reunión. José no llega tarde a su reunión. El tren llegó tarde. Por lo tanto, había taxis en la estación.

Si consideramos,

p = “El tren llega tarde”
q = “hay taxis en la estación”
r = “José llega tarde a su reunión”

El secuente sería $p \wedge \sim q \rightarrow r, \sim r, p \mid\!- q$

Deducción natural

Si consideramos

p = “El oro es un metal”

q = “La plata es un metal”

¿Qué puede decir respecto al secuente $p, q \vdash p \wedge \sim q$?

Deducción natural

Adjunción o introducción de la conjunción

$$\frac{\varphi \quad \psi}{\varphi \land \psi}$$

Eliminación de la conjunción

$$\frac{\varphi \land \psi}{\varphi} \qquad \frac{\varphi \land \psi}{\psi}$$

Deducción natural

Eliminación de la implicancia

$$\frac{\varphi \quad \varphi \rightarrow \psi}{\psi}$$

Modus Tollens

$$\frac{\varphi \rightarrow \psi \quad \sim\psi}{\sim\varphi}$$

Adjunción o introducción de la implicancia

$$\frac{\boxed{\begin{array}{c}\varphi \\ \cdot \\ \cdot \\ \cdot \\ \psi\end{array}}}{\varphi \rightarrow \psi}$$

Deducción natural

Adjunción o introducción de la doble negación

$$\frac{\varphi}{\sim\sim\varphi}$$

Eliminación de la doble negación

$$\frac{\sim\sim\varphi}{\varphi}$$

Deducción natural

Adjunción o introducción de la disyunción

$$\frac{\varphi}{\varphi \lor \psi} \qquad \frac{\psi}{\varphi \lor \psi}$$

Eliminación de la disyunción

$$\frac{\varphi \lor \psi \quad \boxed{\begin{array}{c}\varphi\\ \vdots\\ \chi\end{array}} \quad \boxed{\begin{array}{c}\psi\\ \vdots\\ \chi\end{array}}}{\chi}$$

Deducción natural

Adjunción o introducción de la negación

$$\dfrac{\boxed{\begin{array}{c}\varphi\\ \cdot\\ \cdot\\ \cdot\\ \bot\end{array}}}{\sim\varphi}$$

Eliminación de la negación

$$\dfrac{\varphi \quad \sim\varphi}{\bot}$$

Deducción natural

Demostración por contradicción

$$\frac{\boxed{\begin{array}{c}\sim\varphi \\ \vdots \\ \bot\end{array}}}{\varphi}$$

Eliminación de la contradicción

$$\frac{\bot}{\varphi}$$

Lógica Proposicional - **Sintaxis**

Definimos un conjunto de símbolos (alfabeto):

- Un conjunto P "posiblemente infinito", de variables proposicionales: p, q, r, s
- Constantes: V o F (T o F, 1 ó 0)
- Conectivos lógicos: ~ (¬), ∨ (+), ∧ (*), →, ↔
- Símbolos de puntuación: (,)

"posiblemente infinito" hace alusión a infinito numerable

Lógica Proposicional - **Sintaxis**

Definimos un conjunto de símbolos (alfabeto):

- Un conjunto P “posiblemente infinito”, de variables proposicionales: p, q, r, s
- Constantes: V o F (T o F, 1 ó 0)
- Conectivos lógicos: ~ (¬), ∨ (+), ∧ (*), →, ↔
- Símbolos de puntuación: (,)

Existe una precedencia de operadores: ~, ∨, ∧, →, ↔

“posiblemente infinito” hace alusión a infinito numerable

Lógica Proposicional - **Sintaxis**

A partir de un conjunto fijo *P* de variables, es posible definir un lenguaje proposicional ***L(P)***, que contiene **todas las fórmulas posibles** a través de una definición inductiva.

Así, *L*(*P*) está formado por fórmulas, donde una fórmula es:

- Una constante o elemento de P (fórmulas atómicas).
- Si φ es una fórmula, entonces ~ φ también es una fórmula.
- Si φ y ψ son fórmulas, entonces (φ ○ ψ) es una fórmula (○ representa cualquier conectivo lógico binario).

Definición: Una fórmula bien formada en lógica proposicional (FBF) es aquella que es obtenida usando únicamente las reglas de construcción antes definidas una cantidad finita de veces.

Lógica Proposicional - **Sintaxis**

Definiciones inductivas, como la anterior, son muy frecuentes, tanto que se suelen definir empleando una gramática en Backus Naur Form (BNF). En esa forma, la definición anterior se lee de manera más compacta como:

$$\varphi ::= p \mid (\sim\varphi) \mid (\varphi \land \varphi) \mid (\varphi \lor \varphi) \mid (\varphi \rightarrow \varphi)$$

donde p representa cualquier proposición atómica y cualquier ocurrencia de φ a la derecha de ::= representa cualquier fórmula antes construída.

Lógica Proposicional - **Sintaxis**

Muestre que las fórmulas son FBF:

1. $(p \vee q) \wedge q$
2. $(p \wedge \sim q \sim) \wedge \vee r)$

Lógica Proposicional - **Sintaxis**

Sumemos los primeros 10 números naturales.

Lógica Proposicional - **Sintaxis**

Sumemos los primeros 10 números naturales. no tan difícil.

Lógica Proposicional - **Sintaxis**

Sumemos los primeros 10 números naturales. no tan difícil.

Ahora, los 100 primeros naturales.

Lógica Proposicional - **Sintaxis**

Sumemos los primeros 10 números naturales. no tan difícil.

Ahora, los 100 primeros naturales. un poco más difícil.

Lógica Proposicional - **Sintaxis**

Sumemos los primeros 10 números naturales. no tan difícil.

Ahora, los 100 primeros naturales. un poco más difícil.

y ¿si queremos la suma de los 1000 primeros naturales?

Lógica Proposicional - **Sintaxis**

Recordemos la Inducción Matemática…

Sumemos los primeros 10 números naturales. no tan difícil.

Ahora, los 100 primeros naturales. un poco más difícil.

y ¿si queremos la suma de los 1000 primeros naturales?....

Mediante inducción,

$$1 + 2 + 3 + \ldots + n = \frac{n(n+1)}{2}$$

Lógica Proposicional - **Sintaxis**

La inducción matemática nos permite probar que determinada propiedad es satisfecha por cualquier número natural. Por ejemplo, para el caso anterior, basta definir M(k) para indicar que la propiedad es satisfecha por k. Así, supongamos que conocemos las siguientes propiedades para M:

Caso Base: El número natural 1 satisface M(1).

Paso inductivo: Se asume que la propiedad M(n) es cierta para un determinado número natural n, luego se debe probar que para n+1 se satisface M(n+1), es decir, existe una prueba de que $M(n) \rightarrow M(n+1)$.

Lógica Proposicional - **Sintaxis**

Definición: el principio de inducción matemática dice que, sobre la base de la información anterior, cualquier número natural cumple la propiedad M(n). La suposición de M(n) en el paso inductivo es llamada la Hipótesis de inducción.

Lógica Proposicional - **Sintaxis**

Definición: el principio de inducción matemática dice que, sobre la base de la información anterior, cualquier número natural cumple la propiedad M(n). La suposición de M(n) en el paso inductivo es llamada la Hipótesis de inducción.

Definición: Sea φ una FBF, definimos su altura como la suma entre 1 y el tamaño del camino más largo de su árbol de análisis sintáctico.

Lógica Proposicional - **Sintaxis**

Definición: el principio de inducción matemática dice que, sobre la base de la información anterior, cualquier número natural cumple la propiedad M(n). La suposición de M(n) en el paso inductivo es llamada la Hipótesis de inducción.

Definición: Sea φ una FBF, definimos su altura como la suma entre 1 y el tamaño del camino más largo de su árbol de análisis sintáctico.

Por ejemplo,

Ya que cualquier FBF tiene tamaño finito, podemos mostrar declaraciones sobre todas las FBF por Inducción matemática en sus alturas.

> Esta propiedad se conoce habitualmente como **inducción estructural**, técnica de razonamiento muy importante en ciencias de la computación.

Lógica Proposicional - **Sintaxis**

Teorema: para cada FBF, el número de paréntesis izquierdos es igual al número de paréntesis derechos.

¿Demostración?

Lógica Proposicional - **Sintaxis**

Teorema: para cada FBF, el número de paréntesis izquierdos es igual al número de paréntesis derechos.

<u>Demostración</u>

Los pasos a seguir son, usando inducción de curso de valores sobre la altura de la FBF φ, definimos M(n) = "todas las fórmulas de altura n tienen el mismo número de paréntesis izquierdos y derechos". Luego, asumimos M(k) para cada $k < n$, e intentamos probar M(n).

Lógica Proposicional - **Sintaxis**

Teorema: para cada FBF, el número de paréntesis izquierdos es igual al número de paréntesis derechos.

Demostración

Los pasos a seguir son, usando inducción de curso de valores sobre la altura de la FBF φ, definimos **M(n) = "todas las fórmulas de altura n tienen el mismo número de paréntesis izquierdos y derechos".** Luego, asumimos M(k) para cada k < n, e intentamos probar M(n).

Lógica Proposicional - **Sintaxis**

Teorema: para cada FBF, el número de paréntesis izquierdos es igual al número de paréntesis derechos.

Demostración

Los pasos a seguir son, usando inducción de curso de valores sobre la altura de la FBF φ, definimos **M(n) = “todas las fórmulas de altura n tienen el mismo número de paréntesis izquierdos y derechos”.** Luego, **asumimos M(k) para cada k < n**, e intentamos probar M(n).

Lógica Proposicional - **Sintaxis**

Caso base: n = 1. En este caso φ es una proposición atómica, por lo que no hay paréntesis izquierdos ni tampoco paréntesis derechos, es decir, 0 = 0.

Inducción de curso de valores: n > 1. En este caso, el inicio del árbol de análisis de φ debe ser algunos de los conectores ~, $\wedge$, $\vee$, $\rightarrow$, para φ una FBF. En particular y sin pérdida de generalidad, asumimos que es $\rightarrow$ (los otros casos siguen el mismo razonamiento). Luego φ equivale a $(\varphi_1 \rightarrow \varphi_2)$ para φ_1, φ_2 FBF (φ_1, φ_2 son las representaciones lineales de los subárboles izquierdo y derecho de φ). Dado que las alturas de φ_1 y φ_2 son estrictamente menores que n, usando la hipótesis inductiva, podemos concluir que φ_1 tiene el mismo número de paréntesis izquierdos y derechos, y lo mismo para φ_2. Pero en $(\varphi_1 \rightarrow \varphi_2)$ agregamos dos paréntesis más. Así, el número de ocurrencias de paréntesis izquierdos y derechos es el mismo.

Lógica Proposicional - **Sintaxis**

¿Cómo determinar validez?

Lógica Proposicional - **Semántica**

La semántica debe proveer tres cosas:

- Significado de las fórmulas.
- Noción de verdad
- Noción de consecuencia lógica.

Lógica Proposicional - **Semántica**

La semántica debe proveer tres cosas:

- Significado de las fórmulas.
- **Noción de verdad (Verdadero o Falso?)**
- Noción de consecuencia lógica.

Lógica Proposicional - **Semántica**

Ejemplo:

Si tomas el medicamento, entonces te mejorarás.
No estás mejorando.

Por lo tanto, no tomaste el medicamento.

Lógica Proposicional - **Semántica**

Ejemplo:

Si tomas el medicamento, **entonces** te mejorarás.
No estás mejorando.

Por lo tanto, **no** tomaste el medicamento.

Si p **entonces** q
no q

---

no p

Lógica Proposicional - **Noción de verdad**

Definición: una valuación o asignación de verdad es una función

$$\sigma: P \to \{ 0, 1 \}$$

Lógica Proposicional - **Noción de verdad**

Definición: una valuación o asignación de verdad es una función

$$\sigma: P \rightarrow \{ 0, 1 \}$$

Ejemplo:

Si $P = \{p, q\}$, la función σ_1 es una valuación definida por:

$$\sigma_1(p) = 1, \sigma_1(q) = 0$$

Lógica Proposicional - **Noción de verdad**

Definición: una valuación o asignación de verdad es una función

$$\sigma : P \to \{ 0, 1 \}$$

Ejemplo:

Si $P = \{p, q\}$, la función σ_1 es una valuación definida por:

$$\sigma_1(p) = 1, \sigma_1(q) = 0$$

¿Cuántas funciones de valuación distintas existen para el conjunto *P*?

Lógica Proposicional - **Noción de verdad**

Si φ y ψ son fórmulas proposicionales, entonces:

φ	ψ	$\sim\varphi$	$\varphi \wedge \psi$	$\varphi \vee \psi$	$\varphi \rightarrow \psi$	$\varphi \leftrightarrow \psi$
0	0	1	0	0	1	1
0	1	1	0	1	1	0
1	0	0	0	1	0	0
1	1	0	1	1	1	1

Definición: una fórmula ψ es **equivalente** a otra χ ssi σ(ψ) = σ(χ) para toda valuación σ (misma tabla de verdad).

Lógica Proposicional - **CFC**

Definición: un conjunto *C* de conectivos es funcionalmente completo si es posible definir a todos los conectivos estándar en función de los que contiene *C*.

Ejemplo:

$C = \{\sim, \vee, \wedge\}$ es funcionalmente completo.

Más aún,

$C' = \{\sim, \wedge\}$ es funcionalmente completo.

$(p \vee q)$ es equivalente a $\sim(\sim p \wedge \sim q)$

Lógica Proposicional - **Formas normales**

Las formas normales son formas sintácticas estándares que pueden cumplir las fórmulas.

Recordemos dos formas:

- Forma Normal Conjuntiva o **FNC**
- Forma Normal Disyuntiva o **FND**

Lógica Proposicional - **Formas normales**

Definición: un literal es una variable proposicional, o una variable proposicional negada o una constante V o F.

Definición: una **cláusula** es una disyunción de literales, es decir, de la forma:

$$l_1 \lor l_2 \lor l_3 \lor \ldots \lor l_k$$

Definición: una **cláusula dual** es una conjunción de literales, es decir, de la forma:

$$l_1 \land l_2 \land l_3 \land \ldots \land l_k$$

Lógica Proposicional - **Formas normales**

Definición: una fórmula en **Forma Normal Conjuntiva** (FNC) es una **conjunción** de cláusulas. Si cada c_i es una cláusula, entonces la FNC es de la forma:

$$c_1 \land c_2 \land c_3 \land \dots \land c_n$$

Ejemplo: $(p \lor \sim q \lor s) \land (p \lor s) \land p$

Lógica Proposicional - **Formas normales**

Definición: una fórmula en **Forma Normal Disyuntiva** (FND) es una **disyunción** de cláusulas duales. Si cada c'_i es una cláusula dual, entonces la FND es de la forma:

$$c'_1 \lor c'_2 \lor c'_3 \lor \dots \lor c'_n$$

Ejemplo: $(p \land q) \lor q \lor (\sim p \land s)$

Lógica Proposicional - **Formas normales**

Teorema: Toda fórmula es equivalente a una fórmula en **FND**.

Teorema: Toda fórmula es equivalente a una fórmula en **FNC**.

Lógica Proposicional - **Formas normales**

Teorema: Toda fórmula es equivalente a una fórmula en **FND**.

Teorema: Toda fórmula es equivalente a una fórmula en **FNC**.

$\{\sim, \wedge, \vee\}$ es funcionalmente completo

Lógica Proposicional - **Fórmula satisfacible**

Definición: una fórmula es satisfacible si es verdadera para alguna valuación σ.

Ejemplo: $\sim p \wedge (p \vee q)$

p	**q**	**~p**	**p ∨ q**	**~p ∧ (p ∨ q)**
0	0	1	0	0
0	1	1	1	1
1	0	0	1	0
1	1	0	1	0

Lógica Proposicional - **Fórmula satisfacible**

Definición: una fórmula es satisfacible si es verdadera para alguna valuación σ.

Ejemplo: $\sim p \land (p \lor q)$

p	q	~p	p ∨ q	~p ∧ (p ∨ q)
0	0	1	0	0
0	1	1	1	1
1	0	0	1	0
1	1	0	1	0

Lógica Proposicional - **Fórmula satisfacible**

Definición: un conjunto de fórmulas Σ es satisfacible **si existe al menos una valuación** σ que hace verdaderas a todas las fórmulas de Σ.

Notación: $\sigma \models \Sigma$

Un conjunto de fórmulas que no se puede satisfacer (insatisfacible) se le conoce como **inconsistente**.

Lógica Proposicional - **Fórmulas válidas**

Definición: una fórmula es válida si es satisfacible para toda valuación σ. También son conocidas como **Tautologías**.

Ejemplo: para la fórmula $p \vee \sim p$

p	**~p**	**p ⋁ ~p**
0	1	1
1	0	1

Lógica Proposicional - **Fórmulas válidas**

Contradicción: proposición que en todos los casos posibles de su tabla de verdad su valor siempre es F.

Contingencia: proposición que puede ser verdadera o falsa.

Lógica Proposicional - **Fórmulas válidas**

Contradicción: proposición que en todos los casos posibles de su tabla de verdad su valor siempre es F.

Contingencia: proposición que puede ser verdadera o falsa.

Si toma el medicamento, se recuperará
ud. no se está tomando el medicamento

Por lo tanto, ud. no se recuperará

Ejercicios

Determine si las siguientes oraciones proposicionales son tautología, contradicción o satisfacible.

a) $(p \rightarrow (q \rightarrow p))$

b) $((p \leftrightarrow \sim q) \vee q)$

c) $((p \rightarrow q) \rightarrow (\sim(q \rightarrow p)))$

d) $\sim(((p \rightarrow q) \rightarrow p) \rightarrow p)$

Lógica Proposicional - **Satisfacibilidad**

- **Una fórmula φ** es satisfacible si es verdadera para alguna valuación σ.

$$\sigma \models \varphi$$

Lógica Proposicional - **Satisfacibilidad**

- **Una fórmula φ** es satisfacible si es verdadera para alguna valuación σ.

$$\sigma \models \varphi$$

- **Un conjunto** de **fórmulas Σ** es satisfacible si existe al menos una valuación σ que hace verdaderas a las fórmulas de Σ.

$$\sigma \models \Sigma$$

Lógica Proposicional - **Satisfacibilidad**

- **Una fórmula φ** es satisfacible si es verdadera para alguna valuación σ.

$$\sigma \models \varphi$$

- **Un conjunto** de **fórmulas Σ** es satisfacible si existe al menos una valuación σ que hace verdaderas a las fórmulas de Σ.

$$\sigma \models \Sigma$$

- Una fórmula **φ** es válida o **tautología** si es satisfacible para **toda** valuación σ.

$$\models \varphi$$

Noción de verdad...

Lógica Proposicional - **Inferencia**

Inferencia válida

Una inferencia es válida **si y sólo si** en todos los casos en los que todas las premisas son verdaderas, la conclusión también es verdadera.

$$\frac{\varphi_1, \dots, \varphi_n}{\psi}$$

Lógica Proposicional - **Inferencia**

Inferencia válida

Una inferencia es válida **si y sólo si** en todos los casos en los que todas las premisas son verdaderas, la conclusión también es verdadera.

$$\frac{\varphi_1, \ldots, \varphi_n}{\psi}$$

ψ es una consecuencia lógica de $\Sigma = \{\varphi_1, \ldots, \varphi_n\}$

$$\varphi_1, \ldots, \varphi_n \models \psi$$

$$\Sigma \models \psi.$$

Lógica Proposicional - **Idea deducción**

Supongamos que tenemos la siguiente fórmula en FNC:

$$\boxed{(p \vee q \vee \sim r) \wedge (\sim q \vee s \vee \sim v)}$$

y que σ es una valuación que la hace verdadera.

- Si $\sigma \models q$ .... $\sigma \models s \vee \sim v$
- Si $\sigma \not\models q$ .... $\sigma \models p \vee \sim r$

Por lo tanto, podemos concluir que $\sigma \models (p \vee \sim r \vee s \vee \sim v)$.

Lógica Proposicional - **Idea deducción**

Supongamos que tenemos la siguiente fórmula en FNC:

$$\boxed{(p \vee q \vee \sim r) \wedge (\sim q \vee s \vee \sim v)}$$

y que σ es una valuación que la hace verdadera.

- Si $\sigma \models q \ldots \sigma \models s \vee \sim v$
- Si $\sigma \not\models q \ldots \sigma \models p \vee \sim r$

Por lo tanto, podemos concluir que $\sigma \models (p \vee \sim r \vee s \vee \sim v)$.

Hemos conseguido eliminar un literal y su negado, para obtener una cláusula con una proposición menos.

Lógica Proposicional - **Deducción**

Teorema de deducción

Sea $\Sigma \subseteq L(P)$, entonces

$$\boxed{\Sigma \models (\varphi \rightarrow \psi) \text{ si y sólo si } \Sigma \cup \{\varphi\} \models \psi}$$

Lógica Proposicional - **DPR**

Demostración Por Resolución (**DPR**)

$$\Sigma \models \psi$$

Está basado en la reducción entre **consistencia** y **consecuencia** lógica

$$\Sigma \models \psi \Leftrightarrow \Sigma \cup \{\sim\psi\} \text{ es inconsistente } (\Sigma \models \square)$$

$\square$: Cláusula vacía (sin literales), por lo tanto no satisfacible

Lógica Proposicional - **DPR**

Método para demostrar $\Sigma \models \psi$

- Transformar $\Sigma \cup \{\sim\psi\}$ a FNC, con un conjunto de cláusulas $C = \{C_1, ..., C_n\}$.

- Mientras $\square \notin C$ y existen $C_i, C_j \in C$ para aplicar regla de resolución:
  - Aplicar regla de resolución a C_i, C_j, generando *C'*.
  - Hacer **C** := *C* U {*C'*}.

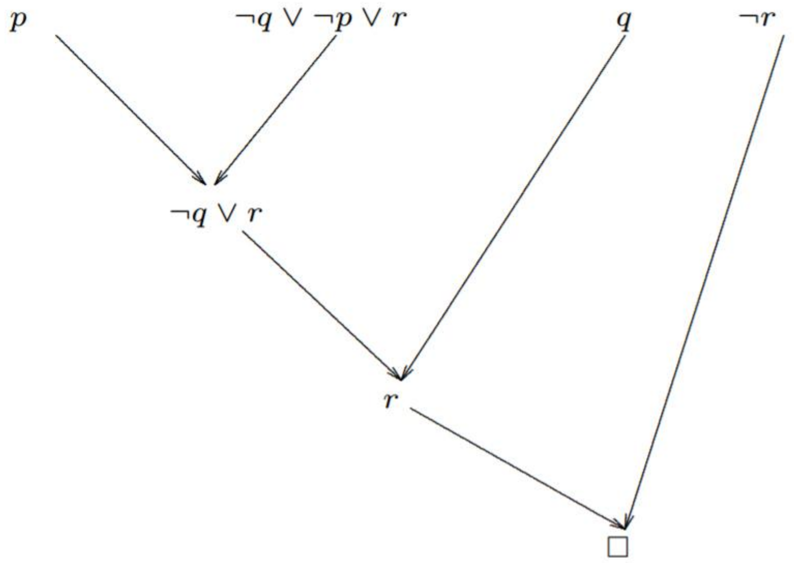
Si $\square \in C$, entonces $\Sigma \models \psi$

Ejemplo

Sea $\Sigma = \{p, q \rightarrow (p \rightarrow r)\}$, demostrar $\Sigma \models (q \rightarrow r)$

Ejemplo

Sea $\Sigma = \{p, q \rightarrow (p \rightarrow r)\}$, demostrar $\Sigma \models (q \rightarrow r)$



Lógica Proposicional - **Inferencia simbólica**

- Una ***demostración*** es una secuencia finita de fórmulas (válidas) en la cual cada fórmula es un axioma, o bien ha sido derivada a partir de fórmulas anteriores mediante una regla de inferencia.

- Un ***teorema*** es una fórmula derivada de una demostración.

- Un ***sistema axiomático*** es un conjunto de axiomas y reglas de inferencia.
  - Un sistema axiomático es ***correcto*** para una lógica dada si todo teorema es válido en la lógica.
  - Un sistema axiomático es ***completo*** para una lógica dada si toda fórmula válida de la lógica es un teorema.

Lógica Proposicional

- **Modelar** el proceso de **razonamiento** a través del descubrimiento de **hechos**
- **Formalizar** la noción de **demostración**

Lógica Proposicional

Cómo expresamos...

Algunas personas son mujeres
Héctor **es persona**

Héctor **es mujer**

Lógica Proposicional

Cómo expresamos...

Algunas personas son mujeres
Héctor **es persona**

Héctor **es mujer**

y cómo demostramos que....

cualquier número natural es par o impar

Lógica Proposicional

- El poder expresivo de la lógica proposicional es limitado:
    - LP **no permite** referirse fácilmente a todos los elementos de un dominio.
    - Si el dominio es infinito no numerable, simplemente no se puede expresar el conocimiento acerca de todos los individuos.

    entonces…..

Lógica de Primer Orden

UdeSantiago de Chile

Lógica de Primer Orden - LPO

Mientras que

- LP **asume que el mundo tiene sólo** hechos.
- LPO **asume que el mundo tiene**:
    - **Objetos**: personas, casas, números, animales, profesores, instituciones, …
    - **Predicados** (o relaciones): hermano de, mayor que, dentro de, de color, es dueño de, hijo de, padre de, …
    - **Funciones**: sucesor de, raíz cuadrada de, segundo tiempo de, mejor amigo de, ...

Lógica de Primer Orden - LPO

- “uno más dos es igual a tres”
  - Objetos: uno, dos, tres
  - Relaciones: es igual a
  - Funciones: más

Lógica de Primer Orden - LPO

- "**uno** más **dos** es igual a **tres**"
  - Objetos: **uno, dos, tres**
  - Relaciones: es igual a
  - Funciones: más

Lógica de Primer Orden - LPO

- “uno más dos **es igual a** tres”
  - Objetos: uno, dos, tres
  - Relaciones: **es igual a**
  - Funciones: más

Lógica de Primer Orden - LPO

- “uno **más** dos es igual a tres”
  - Objetos: uno, dos, tres
  - Relaciones: es igual a
  - Funciones: **más**

Lógica de Primer Orden - LPO

- “uno más dos es igual a tres”
  - <u>Objetos</u>: uno, dos, tres
  - <u>Relaciones</u>: es igual a
  - <u>Funciones</u>: más
- “los alumnos de lógica tienen sueño”
  - <u>Objetos</u>:

Lógica de Primer Orden - LPO

- "uno más dos es igual a tres"
  - Objetos: uno, dos, tres
  - Relaciones: es igual a
  - Funciones: más
- "los alumnos de lógica tienen sueño"
  - Objetos: alumnos
  - Relaciones:

Lógica de Primer Orden - LPO

- “uno más dos es igual a tres”
  - Objetos: uno, dos, tres
  - Relaciones: es igual a
  - Funciones: más
- “los alumnos de lógica tienen sueño”
  - Objetos: alumnos
  - Relaciones: de lógica, tienen sueño
  - Funciones:

Lógica de Primer Orden - LPO

- "uno más dos es igual a tres"
  - Objetos: uno, dos, tres
  - Relaciones: es igual a
  - Funciones: más
- "los alumnos de lógica tienen sueño"
  - Objetos: alumnos
  - Relaciones: de lógica, tienen sueño
  - Funciones: -------

Lógica de Primer Orden - LPO

- "uno más dos es igual a tres"
  - Objetos: uno, dos, tres
  - Relaciones: es igual a
  - Funciones: más
- "los alumnos de lógica tienen sueño"
  - Objetos: alumnos
  - Relaciones: de lógica, tienen sueño
  - Funciones: -------
- "Homero es padre de Bart y esposo de Marge"
  - Objetos:

Lógica de Primer Orden - LPO

- “uno más dos es igual a tres”
  - Objetos: uno, dos, tres
  - Relaciones: es igual a
  - Funciones: más
- “los alumnos de lógica tienen sueño”
  - Objetos: alumnos
  - Relaciones: de lógica, tienen sueño
  - Funciones: -------
- “Homero es padre de Bart y esposo de Marge”
  - Objetos: Homero, Bart, Marge
  - Relaciones:

Lógica de Primer Orden - LPO

- "uno más dos es igual a tres"
  - Objetos: uno, dos, tres
  - Relaciones: es igual a
  - Funciones: más
- "los alumnos de lógica tienen sueño"
  - Objetos: alumnos
  - Relaciones: de lógica, tienen sueño
  - Funciones: -------
- "Homero es padre de Bart y esposo de Marge"
  - Objetos: Homero, Bart, Marge
  - Relaciones: es padre de, esposo de
  - Funciones:

Lógica de Primer Orden - LPO

- “uno más dos es igual a tres”
  - Objetos: uno, dos, tres
  - Relaciones: es igual a
  - Funciones: más
- “los alumnos de lógica tienen sueño”
  - Objetos: alumnos
  - Relaciones: de lógica, tienen sueño
  - Funciones: -------
- “Homero es padre de Bart y esposo de Marge”
  - Objetos: Homero, Bart, Marge
  - Relaciones: es padre de, esposo de
  - Funciones: -------

Lógica de Primer Orden - LPO

- "uno más dos es igual a tres"
  - Objetos: uno, dos, tres
  - Relaciones: es igual a
  - Funciones: más
- "los alumnos de lógica tienen sueño"
  - Objetos: alumnos
  - Relaciones: de lógica, tienen sueño
  - Funciones: -------
- "Homero es padre de Bart y esposo de Marge"*
  - Objetos: Homero, Bart, Marge
  - Relaciones: es padre de, esposo de
  - Funciones: -------

*y es un **conectivo lógico**.

Lógica de Primer Orden - LPO

Alfabeto

- Constantes (**C**): persona, casa, Homero, dos, alumnos, etc.
- Predicados o relaciones (**P**): es igual a, es padre de, tienen sueño, pertenece a, etc.
- Funciones (**F**): más, raíz cuadrada, sucesor, etc.
- Variables: x, y, z, etc.
- Conectivos lógicos: $\neg$, $\vee$, $\wedge$, $\rightarrow$, $\leftrightarrow$
- Cuantificadores lógicos: $\forall$, $\exists$
- Relación igualdad: =
- Símbolos de puntuación: (,)

El concepto de ***predicado*** es similar al de ***relación***, con la única salvedad que el primero es filosófico y el segundo es matemático.

Lógica de Primer Orden - LPO

Predicados y funciones

- Los predicados suelen ser mapeos de objetos a un V o F.
- Los predicados y funciones tienen una aridad mayor a 0.
- Las funciones son mapeos de ***muchos*** a ***un*** objeto.

Ejemplo,

P = {GustaComer (·), EsHermanoDe(· , ·)}
F = {Hijo(·)}
C = {Homero, Marge, Bart, Lisa, Maggie}

Lógica de Primer Orden - LPO

- Una fórmula en LPO está definida sobre algunas constantes, funciones y predicados.

- Un vocabulario L (o un conjunto de símbolos) es la unión tres conjuntos:
  - Constantes: $\{c_1, c_2, ..., c_l, ...\}$
  - Funciones: $\{f_1, f_2, ..., f_m, ...\}$
  - Predicados: $\{p_1, p_2, ..., p_m, ...\}$

$$L = \{\{p_1, p_2, ..., p_m, ...\}, \{f_1, f_2, ..., f_m, ...\}, \{c_1, c_2, ..., c_l, ...\}\}$$

Lógica de Primer Orden - LPO

- En los números naturales,
  - Constantes: {0, 1}
  - Funciones: {sucesor, +, *}
  - Relaciones: {<}

  Ejemplo,
    - sucesor es una función unaria: sucesor(·)
    - + y * son funciones binarias: +(· , ·) y *(· , ·)
    - < es una relación binaria: <(· , ·)

- L = { {GustaComer}, {hijo}, {Homero, Marge, Bart, Lisa, Maggie} } un vocabulario donde:
  - GustaComer es un predicado de aridad 1 tal que GustaComer(x) es verdadero si a x le gusta comer.
  - La función hijo(x) representa al padre del objeto x.

Lógica de Primer Orden - LPO

Variables

- Se utilizan en predicados y funciones

Ejemplo: GustaComer (x), EsHermanoDe(x, y), Hijo(z).

- Pueden ser:
  - Ligadas a algún cuantificador $\forall$ o $\exists$.
  - Libres (no ligadas).

Ejemplo: $P(x, y) \land \forall z Q(z)$

Lógica de Primer Orden - LPO

Variables

- Se utilizan en predicados y funciones

Ejemplo: GustaComer (x), EsHermanoDe(x, y), Hijo(z).

- Pueden ser:
  - Ligadas a algún cuantificador $\forall$ o $\exists$.
  - Libres (no ligadas).

Ejemplo: $P(x, y) \land \forall z Q(z)$

➔ La x e y en P(x, y) son libres

Lógica de Primer Orden - LPO

Variables

- Se utilizan en predicados y funciones

Ejemplo: GustaComer (x), EsHermanoDe(x, y), Hijo(z).

- Pueden ser:
  - Ligadas a algún cuantificador $\forall$ o $\exists$.
  - Libres (no ligadas).

Ejemplo: $P(x, y) \land \forall z Q(z)$

→ La x e y en P(x, y) son libres

→ La z de $\forall z Q(z)$ está ligada

Lógica de Primer Orden - LPO

Variables

- Dado que las variables libres no aparecen cuantificadas sobre el dominio, el valor de verdad de la fórmula dependerá del valor dado a las variables libres.

Ejemplo:

- ¿Es cierto que $x < \text{sucesor}(0)$ en $\mathbb{N}$?

Lógica de Primer Orden - LPO

Variables

- Dado que las variables libres no aparecen cuantificadas sobre el dominio, el valor de verdad de la fórmula dependerá del valor dado a las variables libres.

Ejemplo:

- ¿Es cierto que $x < \text{sucesor}(0)$ en $\mathbb{N}$?
  - Si x es 0, entonces es verdadera en $\mathbb{N}$.
  - Si x es 1, entonces es falsa en $\mathbb{N}$.

Lógica de Primer Orden - LPO

Variables

- Dado que las variables libres no aparecen cuantificadas sobre el dominio, el valor de verdad de la fórmula dependerá del valor dado a las variables libres.

Ejemplo:

- ¿Es cierto que x < sucesor(0) en $\mathbb{N}$?
  - Si x es 0, entonces es verdadera en $\mathbb{N}$.
  - Si x es 1, entonces es falsa en $\mathbb{N}$.
- daHambre(p)
  - ¿Qué pasa si p = hamburguesa?

Lógica de Primer Orden - LPO

Variables

- Dado que las variables libres no aparecen cuantificadas sobre el dominio, el valor de verdad de la fórmula dependerá del valor dado a las variables libres.

Ejemplo:

- ¿Es cierto que $x < \text{sucesor}(0)$ en $\mathbb{N}$?
  - Si x es 0, entonces es verdadera en $\mathbb{N}$.
  - Si x es 1, entonces es falsa en $\mathbb{N}$.
- daHambre(p)
  - ¿Qué pasa si p = hamburguesa?
  - ¿Qué pasa si p = **hákarl**?

Lógica de Primer Orden - LPO

Variables

- Dado que las variables libres no aparecen cuantificadas sobre el dominio, el valor de verdad de la fórmula dependerá del valor dado a las variables libres.

Ejemplo:

- ¿Es cierto que x < sucesor(0) en ℕ?
  - Si x es 0, entonces es verdadera en ℕ.
  - Si x es 1, entonces es falsa en ℕ.
- daHambre(p)
  - ¿Qué pasa si p = hamburguesa?
  - ¿Qué pasa si p = **hákarl**?

Lógica de Primer Orden - LPO

Variables

- En LPO, cada variable presenta un ámbito de influencia (comparable con las variables de clase y de método en Java)
- Una variable puede ser libre y ligada a la vez:
  - $P(x) + \forall x\, Q(x)$
    - En este caso, la variable x para P es **libre** y es **ligada** para Q
    - Dado que x para Q es cuantificada universalmente, Q **es verdadera** cuando se cumple para todos los elementos de U.
    - Dado que para P es libre, la fórmula **será verdadera** dependiendo del valor en U asignado para x.

Lógica de Primer Orden - LPO

Variables

- En LPO, cada variable presenta un ámbito de influencia (comparable con las variables de clase y de método en Java)
- Una variable puede ser libre y ligada a la vez:
  - $P(x) + \forall x\, Q(x)$
    - En este caso, la variable x para P es **libre** y es **ligada** para Q
    - **Dado que x para Q es cuantificada universalmente, Q es verdadera cuando se cumple para todos los elementos de U.**
    - **Dado que para P es libre, la fórmula será verdadera dependiendo del valor en U asignado para x.**

- **¿Qué sucede con los dos últimos puntos?**

Lógica de Primer Orden - LPO

Variables

- Dado que son contradictorios...
  ambas variables x deben ser distintas (x es **sólo** una etiqueta).

$$P(\, x \,) + \forall x \; Q(\, x \,)$$

distintas

- Lo anterior lo podemos re-escribir como:

$$P(\, \mathbf{u} \,) + \forall x \; Q(\, x \,)$$

Llamaremos ***renombramiento*** a la re-escritura anterior.

Lógica de Primer Orden - LPO

Variables

$$\forall x\,(P(x) \rightarrow \exists x\,(Q(x) \rightarrow \forall x\, R(x)))$$

distintas

Lógica de Primer Orden - LPO

Términos

- Dado un vocabulario L = {P, F, C}, un **término** se define como:
  - Todas las variables son términos. Ej: x, y , z, …
  - Todas las constantes son términos. Ej: Fulano, Zutano, …
  - Si $t_1, ..., t_n$ son términos y f una función n-aria, entonces $f(t_1, ..., t_n)$ es un término.

    Ej: suma(x, 5), sucesor (sucesor (1)), …

Lógica de Primer Orden - LPO

Fórmulas

- Una fórmula puede ser:
  - Cerrada: si no tiene variables libres.
  - Abierta: si tiene al menos una variable libre.

- También se pueden clasificar en:
  - Fórmulas atómicas
  - Fórmulas bien formadas (contienen a las atómicas)

Lógica de Primer Orden - LPO

Fórmulas atómicas

- Dado un vocabulario, una fórmula atómica tiene la forma de:
  - $P(t_1, ..., t_k)$, donde:
    - P es un predicado k-ario.
    - $t_1, ..., t_k$ son términos.
- Las fórmulas atómicas nos ayudan a expresar proposiciones de LP (hechos):
  - EsVecinoDe(Pedro, hijo(Pablo))

Lógica de Primer Orden - LPO

Fórmulas atómicas

- Dado un vocabulario, una fórmula atómica tiene la forma de:
  - $P(t_1, ..., t_k)$, donde:
    - P es un predicado k-ario.
    - $t_1, ..., t_k$ son términos.
- Las fórmulas atómicas nos ayudan a expresar proposiciones de LP (hechos):
  - EsVecinoDe(Pedro, hijo(Pablo))

Esta fórmula puede leerse como: “Pedro es vecino del hijo de Pablo”.

Lógica de Primer Orden - LPO

Fórmulas bien formadas

- Una lista de símbolos es una fórmula bien formada (fbf) ssi se puede aplicar un número finito de veces las reglas:
  - Las fórmulas atómicas son fórmulas
  - Si φ es una fórmula, entonces $\neg\varphi$ es una fórmula
  - Si φ y ψ son fórmulas, entonces $(\varphi + \psi)$, $(\varphi * \psi)$, $(\varphi \rightarrow \psi)$ y $(\varphi \leftrightarrow \psi)$ son fórmulas.
  - Si φ es una fórmula y x es una variable, entonces $(\forall x\, \varphi)$ y $(\exists x\, \varphi)$ son fórmulas.

Lógica de Primer Orden - LPO

Ejemplo:

- "Todos los que estudian lógica son inteligentes"

$$\forall\ x\ (\ \text{Estudia}\ (\ x\ ,\ \text{lógica}\ )\rightarrow \text{inteligente}\ (\ x\ ))$$

- "Todos los que tengan 3,94 de promedio reprobarán lógica"

$$\forall\ x\ (\ \text{Tiene Promedio}\ (\ x\ ,\ 3{,}94\ )\rightarrow \text{RepruebaLógica}\ (\ x\ ))$$

- "Todos tienen una madre"

$$\forall\ x\ \exists\ y\ \text{Madre}\ (\ y\ ,\ x\ )$$

Lógica de Primer Orden - LPO

Algunas equivalencias

$$
\begin{array}{ccc}
\neg\forall x\, \varphi x & \text{equivale a} & \exists x\, \neg\varphi x \\
\neg\exists x\, \varphi x & \text{equivale a} & \forall x\, \neg\varphi x \\
\forall x\, \varphi x & \text{equivale a} & \neg\exists x\, \neg\varphi x \\
\exists x\, \varphi x & \text{equivale a} & \neg\forall x\, \neg\varphi x \\
\neg\forall x\, (\varphi x \rightarrow \psi x) & \text{equivale a} & \exists x\, \neg(\varphi x \rightarrow \psi x) \\
 & \text{equivale a} & \exists x\, (\varphi x \wedge \neg\psi) \\
\forall x\, (\varphi x \rightarrow \neg\psi x) & \text{equivale a} & \neg\exists x\, \neg(\varphi x \rightarrow \neg\psi x) \\
 & \text{equivale a} & \neg\exists x\, (\varphi x \wedge \psi x) \\
\forall x\, (\varphi x \wedge \psi x) & \text{equivale a} & \forall x\, \varphi x \wedge \forall x\, \psi x \\
\exists x\, (\varphi x \vee \psi x) & \text{equivale a} & \exists x\, \varphi x \vee \exists x\, \psi x
\end{array}
$$

Lógica de Primer Orden - LPO

Formas normales

Lógica de Primer Orden - LPO

Formas normales

- **Forma Normal Conjuntiva** (FNC)

Lógica de Primer Orden - LPO

Formas normales

- **Forma Normal Conjuntiva** (FNC)

- **Forma Normal Disyuntiva** (FND)

Lógica de Primer Orden - LPO

Formas normales

- **Forma Normal Conjuntiva** (FNC)

$$EsPar\ (mult(x, y)) \land (\neg EsImpar\ (x) \lor \neg EsImpar\ (y))$$

- **Forma Normal Disyuntiva** (FND)

Lógica de Primer Orden - LPO

Formas normales

- **Forma Normal Conjuntiva** (FNC)

$$EsPar(mult(x, y)) \land (\neg EsImpar(x) \lor \neg EsImpar(y))$$

- **Forma Normal Disyuntiva** (FND)

$$\neg EsPar(mult(x, y)) \lor (EsImpar(x) \land EsImpar(y))$$

Lógica de Primer Orden - LPO

Formas normales

- **Forma Normal Conjuntiva** (FNC)

$$EsPar(mult(x, y)) \land (\neg EsImpar(x) \lor \neg EsImpar(y))$$

- **Forma Normal Disyuntiva** (FND)

$$\neg EsPar(mult(x, y)) \lor (EsImpar(x) \land EsImpar(y))$$

- **Forma Normal Rectificada** (FNR)

Lógica de Primer Orden - LPO

Formas normales

- **Forma Normal Conjuntiva** (FNC)

$$EsPar\,(mult(x, y)) \wedge (\neg EsImpar\,(x) \vee \neg EsImpar\,(y))$$

- **Forma Normal Disyuntiva** (FND)

$$\neg EsPar\,(mult(x, y)) \vee (EsImpar\,(x) \wedge EsImpar\,(y))$$

- **Forma Normal Rectificada** (FNR)
  - Ninguna variable aparece libre y ligada a la vez.
  - Cada cuantificador actúa sobre una variable distinta.

Lógica de Primer Orden - LPO

Formas normales

- **Forma Normal Rectificada** (FNR)
  - Ninguna variable aparece libre y ligada a la vez.
  - Cada cuantificador actúa sobre una variable distinta.

$$\forall x(P(x) \rightarrow \exists x(Q(x) \rightarrow \forall x R(x)))$$

$$\forall x \forall y\, (P(x, y) \leftrightarrow (\neg Q(x) \land \exists m \exists f\, (\neg R(m, f))))$$

Lógica de Primer Orden - LPO

Formas normales

- **Forma Normal Rectificada** (FNR)
  - Ninguna variable aparece libre y ligada a la vez.
  - Cada cuantificador actúa sobre una variable distinta.

$$\forall x(P(x) \rightarrow \exists x(Q(x) \rightarrow \forall x R(x)))$$

$$\forall x \forall y\, (P(x, y) \leftrightarrow (\neg Q(x) \land \exists m \exists f\, (\neg R(m, f))))$$

Lógica de Primer Orden - LPO

Formas normales

- **Forma Normal Rectificada** (FNR)
  - Ninguna variable aparece libre y ligada a la vez.
  - Cada cuantificador actúa sobre una variable distinta.

$$\forall x(P(x) \rightarrow \exists x(Q(x) \rightarrow \forall x R(x)))$$

$$\forall x \forall y\, (P(x, y) \leftrightarrow (\neg Q(x) \land \exists m \exists f\, (\neg R(m, f))))$$

Lógica de Primer Orden - LPO

Formas normales

Lógica de Primer Orden - LPO

Formas normales... hay más!

- **Forma Normal Prenex** (FNP)
  - Los cuantificadores están sólo al comienzo.

$$Q_1 x_1 \ldots Q_n x_n \psi$$

donde $Q_i \in \{\exists, \forall\}$ y ψ no tiene cuantificadores.

Ejemplo:

$$\forall x P(x) \land \exists z R(z)$$
$$\exists y \forall x [P(x) \land Q(y)]$$

Lógica de Primer Orden - LPO

Formas normales... hay más!

- **Forma Normal Prenex** (FNP)
  - Los cuantificadores están sólo al comienzo.

$$Q_1 x_1 \ldots Q_n x_n \psi$$

donde $Q_i \in \{\exists, \forall\}$ y ψ no tiene cuantificadores.

Ejemplo:

$$\forall x P(x) \land \exists z R(z)$$

$$\exists y \forall x [P(x) \land Q(y)]$$

Lógica de Primer Orden - LPO

Formas normales... hay más!

- **Forma Normal Prenex** (FNP)
  - Los cuantificadores están sólo al comienzo.

$$Q_1 x_1 \ldots Q_n x_n \psi$$

donde $Q_i \in \{\exists, \forall\}$ y ψ no tiene cuantificadores.

Ejemplo:

$$\forall x P(x) \land \exists z R(z)$$
$$\exists y \forall x [P(x) \land Q(y)]$$

Lógica de Primer Orden - LPO

Formas normales... hay más!

- **Forma Normal Prenex** (FNP)
  - Los cuantificadores están sólo al comienzo.

$$Q_1 x_1 \ldots Q_n x_n \psi$$

donde $Q_i \in \{\exists, \forall\}$ y ψ no tiene cuantificadores.

Ejemplo:

$$\forall x P(x) \land \exists z R(z)$$
$$\exists y \forall x [P(x) \land Q(y)]$$

Teorema

Cualquier fórmula de LPO es **equivalente** a una fórmula en FNP.

Lógica de Primer Orden - LPO

Formas normales... hay más!

- **Forma Normal de Skolem** (FNS)
  - como FNP pero sin existenciales.

$$\forall x_1 \ldots \forall x_n \, \psi$$

donde ψ no tiene cuantificadores.

Ejemplo:

$$\forall x P(x) \land \forall y Q(y)$$
$$\forall x \forall y \forall z [P(x) \land (R(y) \lor Q(z))]$$

Lógica de Primer Orden - LPO

Formas normales... hay más!

- **Forma Normal de Skolem** (FNS)
  - como FNP pero sin existenciales.

$$\forall x_1 \ldots \forall x_n \, \psi$$

donde ψ no tiene cuantificadores.

Ejemplo:

$$\forall x P(x) \land \forall y Q(y)$$

$$\forall x \forall y \forall z [P(x) \land (R(y) \lor Q(z))]$$

Lógica de Primer Orden - LPO

Formas normales… hay más!

- **Forma Normal de Skolem** (FNS)
  - como FNP pero sin existenciales.

$$\forall x_1 \ldots \forall x_n \, \psi$$

donde ψ no tiene cuantificadores.

Ejemplo:

$$\forall x P(x) \wedge \forall y Q(y)$$
$$\forall x \forall y \forall z [P(x) \wedge (R(y) \vee Q(z))]$$

LPO - **Skolemización**

LPO - **Skolemización**

Eliminación de los cuantificadores existenciales

LPO - **Skolemización**

Eliminación de los cuantificadores existenciales

1. Los existenciales se eliminan de izquierda a derecha.
2. $\exists x \ldots \varphi(x) \ldots \rightarrow \ldots \varphi(c) \ldots$, donde c es una nueva constante.
3. $\forall x_1 \ldots \forall x_{k-1} \exists x_k \ldots \varphi(x_k) \ldots \rightarrow \forall x_1 \ldots \forall x_{k-1} \ldots \varphi(f(x_1, \ldots, x_{k-1})) \ldots$, donde f es una nueva función de Skolem.

LPO - **Skolemización**

Eliminación de los cuantificadores existenciales

1. Los existenciales se eliminan de izquierda a derecha.
2. $\exists x ... \varphi(x) ... \rightarrow ... \varphi(c) ...$, donde c es una nueva constante.
3. $\forall x_1 ... \forall x_{k-1} \exists x_k ... \varphi(x_k) ... \rightarrow \forall x_1 ... \forall x_{k-1} ... \varphi(f(x_1, ..., x_{k-1})) ...$, donde f es una nueva función de Skolem.

Ejemplo 1:

$$\forall x \forall y \exists z \forall w\, S(x, y, z, w)$$

LPO - **Skolemización**

Eliminación de los cuantificadores existenciales

1. Los existenciales se eliminan de izquierda a derecha.
2. $\exists x \ldots \varphi(x) \ldots \rightarrow \ldots \varphi(c) \ldots$, donde c es una nueva constante.
3. $\forall x_1 \ldots \forall x_{k-1} \exists x_k \ldots \varphi(x_k) \ldots \rightarrow \forall x_1 \ldots \forall x_{k-1} \ldots \varphi(f(x_1, \ldots, x_{k-1})) \ldots$, donde f es una nueva función de Skolem.

Ejemplo 1:

$$\forall x \forall y \exists z \forall w\, S(x, y, z, w)$$

LPO - **Skolemización**

Eliminación de los cuantificadores existenciales

1. Los existenciales se eliminan de izquierda a derecha.
2. $\exists x ...\varphi(x)... \rightarrow ...\varphi(c)...$, donde c es una nueva constante.
3. $\forall x_1 ... \forall x_{k-1} \exists x_k ...\varphi(x_k)... \rightarrow \forall x_1 ... \forall x_{k-1} ...\varphi(f(x_1, ..., x_{k-1}))...$, donde f es una nueva función de Skolem.

Ejemplo 1:

$$\forall x \forall y \exists z \forall w\, S(x, y, z, w)$$

$$\downarrow$$

$$\forall x \forall y \forall w\, S(x, y, f(x, y), w)$$

LPO - **Skolemización**

Eliminación de los cuantificadores existenciales

1. Los existenciales se eliminan de izquierda a derecha.
2. $\exists x ... \varphi(x) ... \rightarrow ... \varphi(c) ...$, donde c es una nueva constante.
3. $\forall x_1 ... \forall x_{k-1} \exists x_k ... \varphi(x_k) ... \rightarrow \forall x_1 ... \forall x_{k-1} ... \varphi(f(x_1, ..., x_{k-1})) ...$, donde f es una nueva función de Skolem.

Ejemplo 2:

$$\exists w \forall x \exists y \forall z \exists u (P(x, y) + Q(y, z) + R(u, w))$$

LPO - **Skolemización**

Eliminación de los cuantificadores existenciales

1. Los existenciales se eliminan de izquierda a derecha.
2. $\exists x ... \varphi(x) ... \rightarrow ... \varphi(c) ...$, donde c es una nueva constante.
3. $\forall x_1 ... \forall x_{k-1} \exists x_k ... \varphi(x_k) ... \rightarrow \forall x_1 ... \forall x_{k-1} ... \varphi(f(x_1, ..., x_{k-1})) ...$, donde f es una nueva función de Skolem.

Ejemplo 2:

$$\exists w \forall x \exists y \forall z \exists u (P(x, y) + Q(y, z) + R(u, w))$$

LPO - **Skolemización**

Eliminación de los cuantificadores existenciales

1. Los existenciales se eliminan de izquierda a derecha.
2. $\exists x \ldots \varphi(x) \ldots \rightarrow \ldots \varphi(c) \ldots$, donde c es una nueva constante.
3. $\forall x_1 \ldots \forall x_{k-1} \exists x_k \ldots \varphi(x_k) \ldots \rightarrow \forall x_1 \ldots \forall x_{k-1} \ldots \varphi(f(x_1, \ldots, x_{k-1})) \ldots$, donde f es una nueva función de Skolem.

Ejemplo 2:

$$\exists w \forall x \exists y \forall z \exists u (P(x, y) + Q(y, z) + R(u, w))$$

$$\downarrow$$

$$\forall x \forall z (P(x, f(x)) + Q(f(x), z) + R(g(x, z), h(C)))$$

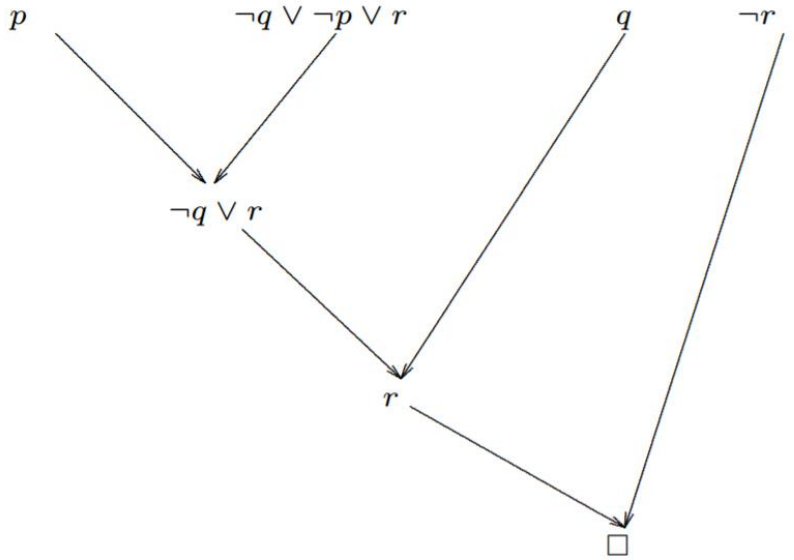
LPO - **Formas normales**

- **Forma Normal Conjuntiva** (FNC)
- **Forma Normal Disyuntiva** (FND)
- **Forma Normal Rectificada** (FNR)
  - Ninguna variable aparece libre y ligada a la vez.
  - Cada cuantificador actúa sobre una variable distinta.
- **Forma Normal Prenex** (FNP)
  - cuantificadores están sólo al comienzo.
- **Forma Normal de Skolem** (FNS)
  - como FNP pero sin existenciales.

LPO - **Método de Resolución**

En LP...

Sea $\Sigma = \{p, q \rightarrow (p \rightarrow r)\}$, demostrar $\Sigma \models (q \rightarrow r)$



LPO - **Método de Resolución**

y en LPO:

LPO - **Método de Resolución**

y en LPO:

Necesitamos algo más...

LPO - **Sistemas de derivación**

- Al igual que en la lógica proposicional, en LPO existen conjuntos de axiomas y reglas de inferencia que permiten derivar cualquier fórmula válida.

- Un teorema es una fórmula que puede ser derivada en un número finito de pasos siguiendo los pasos de un sistema de derivación.

LPO - **Sistemas de derivación**

- Al igual que en la lógica proposicional, en LPO existen conjuntos de axiomas y reglas de inferencia que permiten derivar cualquier fórmula válida.

- Un teorema es una fórmula que puede ser derivada en un número finito de pasos siguiendo los pasos de un sistema de derivación.

- Un sistema de derivación es **correcto** si **todo teorema es una fórmula válida**.

LPO - **Sistemas de derivación**

- Al igual que en la lógica proposicional, en LPO existen conjuntos de axiomas y reglas de inferencia que permiten derivar cualquier fórmula válida.

- Un teorema es una fórmula que puede ser derivada en un número finito de pasos siguiendo los pasos de un sistema de derivación.

- Un sistema de derivación es **correcto** si **todo teorema es una fórmula válida**.

- Un sistema de derivación es **completo** si **toda fórmula válida es un teorema**.

¿Consecuencia lógica?

LPO - **Consecuencia lógica**

"Los chilenos pagan las cuentas a última hora"
"Carlos es chileno"

"Carlos paga las cuentas a última hora"

LPO - **Consecuencia lógica**

“Los chilenos pagan las cuentas a última hora”
“Carlos es chileno”

“Carlos paga las cuentas a última hora”

PREMISAS

LPO - **Consecuencia lógica**

"Los chilenos pagan las cuentas a última hora"
"Carlos es chileno"

"Carlos paga las cuentas a última hora"

CONSECUENCIA

LPO - **Consecuencia lógica**

"Los chilenos pagan las cuentas a última hora"
"Carlos es chileno"

"Carlos paga las cuentas a última hora"

La **consecuencia lógica** es la **relación** que conecta una afirmación (φ) o un conjunto de afirmaciones (Σ) con aquello que está lógicamente implicado por la afirmación o el conjunto de afirmaciones.

LPO - **Consecuencia lógica**

¿Es válido lo siguiente en LP?

$$\frac{(p + q) * (\neg q + r)}{(p + r)}$$

LPO - **Consecuencia lógica**

truth table	$(p \lor q) \land (\neg q \lor r) \Rightarrow p \lor r$

p	q	r	$(p \lor q) \land (\neg q \lor r) \Rightarrow p \lor r$
T	T	T	T
T	T	F	T
T	F	T	T
T	F	F	T
F	T	T	T
F	T	F	T
F	F	T	T
F	F	F	T

WolframAlpha computational knowledge engine

LPO - **Consecuencia lógica**

¿Es válido lo siguiente en LP?

$$\frac{(p + q) * (\neg q + r)}{(p + r)}$$

LPO - **Consecuencia lógica**

¿Es válido lo siguiente en LP?

$$\frac{(p + q) * (\neg q + r)}{(p + r)}$$

LPO - **Consecuencia lógica**

¿Es válido lo siguiente en LP?

$$\frac{(p + q) * (\neg q + r)}{(p + r)}$$

Ahora veamos el caso para LPO:

$$\frac{\forall x\ \forall y\ ((P(x) + Q(y)) * (\neg Q(y) + R(x)))}{\forall x\ \forall y\ (P(x) + R(x))}$$

LPO - **Consecuencia lógica**

¿Es válido lo siguiente en LP?

$$\frac{(p + q) * (\neg q + r)}{(p + r)}$$

Ahora veamos el caso para LPO:

$$\frac{\forall x\ \forall y\ ((P(x) + Q(y)) * (\neg Q(y) + R(x)))}{\forall x\ \forall y\ (P(x) + R(x))}$$

LPO - **Consecuencia lógica**

y con....

$$\frac{\forall x\ \forall y\ ((P(x) + Q(Homero)) * (\neg Q(y) + R(x)))}{\forall x\ \forall y\ (\_\_\_\_\_\_)}$$

LPO - **Consecuencia lógica**

y con....

$$\forall x \; \forall y \; ((P(x) + Q(Homero)) * (\neg Q(y) + R(x)))$$

$$\overline{\qquad\qquad\qquad\qquad\qquad\qquad}$$

$$\forall x \; \forall y \; (????????)$$

LPO - **Consecuencia lógica**

y con....

$$\frac{\forall x\ \forall y\ ((P(x) + Q(Homero)) * (\neg Q(y) + R(x)))}{\forall x\ \forall y\ (????????)}$$

Necesitamos la Unificación

LPO - **Consecuencia lógica**

Unificación:

- **Sustitución**: proceso de asignar un valor a una variable, reemplazandola en toda la fórmula.

- Dados los términos s y t
  - t{x/s} es el resultado de reemplazar todas las ocurrencias de x por s en t.
  - A{x/s} es el resultado de reemplazar todas las ocurrencias libres de x por s en A.

LPO - **Consecuencia lógica**

Unificación:

- Ejemplo:
  - Q(y){**y**/**Homero**}

LPO - **Consecuencia lógica**

Unificación:

- Ejemplo:
    - Q(y){**y**/**Homero**} obtenemos Q(Homero)

LPO - **Consecuencia lógica**

Unificación:

- Ejemplo:
    - Q(y){**y**/**Homero**} obtenemos Q(Homero)
    - S(y){**y**/**f(z)**}

LPO - **Consecuencia lógica**

Unificación:

- Ejemplo:
    - Q(y){**y**/**Homero**} obtenemos Q(Homero)
    - S(y){**y**/**f(z)**} obtenemos S(f(z))

LPO - **Consecuencia lógica**

Unificación

- Dos expresiones son unificables si tienen un unificador.
- Diremos que t es una instancia común de t_1 y t_2 si existe una sustitución θ tal que $t = t_1\boldsymbol{\theta} = t_2\boldsymbol{\theta}$.

LPO - **Consecuencia lógica**

Unificación

- Dos expresiones son unificables si tienen un unificador.
- Diremos que t es una instancia común de t_1 y t_2 si existe una sustitución θ tal que $t = t_1\boldsymbol{\theta} = t_2\boldsymbol{\theta}$.

t_1	t_2	Unificador	Inst. común	comentario
$P(x)$	$P(f(y))$	$\{x / f(y)\}$ ó $\{x / f(z), y / z\}$	$P(x)$ ó $P(f(y))$	
$P(a, b)$	$Q(x, b)$	N/A		tienen distinto símbolo de relación
$P(x)$	$P(f(x))$	N/A		x y $f(x)$ no son unificables
$P(x, y)$	$P(y, x)$	$\{x/y\}$ ó $\{y/x\}$ ó $\{x/z, y/z\}$	$P(x,x)$ ó $P(y,y)$ ó $P(z,z)$	
$P(x, f(x))$	$P(a, f(b))$	N/A		dos valores diferentes sustituyen a x

LPO - **Consecuencia lógica**

Unificador Más General (UMG)

- Definición: Un unificador θ es más general que т (denotado como θ > т) si existe otro unificador λ tal que:

$$t_1 \theta \lambda = t_2\, т$$

- Definición: Un unificador θ es el más general (**UMG**) si θ > т para cualquier unificador т aplicable.

LPO - **Consecuencia lógica**

Unificador Más General (UMG)

- Definición: Un unificador θ es más general que τ (denotado como θ > τ) si existe otro unificador λ tal que:

$$t_1 \theta \lambda = t_2 \tau$$

- Definición: Un unificador θ es el más general (**UMG**) si θ > τ para cualquier unificador τ aplicable.

Intuitivamente, el UMG es el unificador que unifica dos expresiones en la menor cantidad de pasos o sustituciones.

LPO - **Consecuencia lógica**

Unificador Más General (UMG)

Ejemplo:

Dados $P(x, y)$ y $P(y, x)$

LPO - **Consecuencia lógica**

Unificador Más General (UMG)

Ejemplo:

Dados P(x, y) y P(y, x)

{x/a, y/a} no es un **UMG**

{y/x} es un **UMG**

LPO - **Consecuencia lógica**

entonces….

$$\forall x\ \forall y\ ((P(x) + Q(y)) * (\neg Q(Homero) + R(x)))$$

LPO - **Consecuencia lógica**

entonces….

$$\forall x\ \forall y\ ((P(x) + Q(y)) * (\neg Q(Homero) + R(x)))$$

$$\Big\downarrow \{\, y / Homero \,\}$$

$$\forall x\ (P(x) + R(x))$$

LPO - **Método de Resolución**

- ¿Cómo demostramos que $\Sigma \models \varphi$?

LPO - **Método de Resolución**

- ¿Cómo demostramos que $\Sigma \models \varphi$?
- Necesitamos un método que nos ayude a esto.

LPO - **Método de Resolución**

- Necesitamos un método que nos ayude a encontrar, de ser posible, el unificador de máxima generalidad.

- Propuesto en 1965 por J. A. Robinson.

Algoritmo de Robinson

Sean E y F dos términos que queremos unificar. Consideramos inicialmente $\sigma_0 = \{\}$ una sustitución vacía, es decir, que no cambia ninguna variable. Dado que vamos a realizar un proceso iterativo, consideramos inicialmente $E_0 = \sigma_0(E)$ y $F_0 = \sigma_0(F)$. En cada iteración k del algoritmo se realizan los siguientes pasos:

1. Si $E_k = F_k$ entonces las cláusulas E y F son unificables y un unificador de máxima generalidad es $\sigma = \sigma_k \circ \cdots \circ \sigma_0$. Además, el término E_k es el término unificado. En este caso el proceso termina aquí.
2. Si $E_k \neq F_k$ entonces se busca el primer par de discordancia entre E_k y F_k. Sea éste D_k.
3. Si D_k contiene una variable y un término (pueden ser dos variables y una de ellas hace de término) pasamos al siguiente paso. En otro caso los términos no son unificables y terminamos el proceso.

Algoritmo de Robinson

4. Si la variable aparece en el término se produce un ***occur check*** por lo que E y F no unifican y terminamos. Si esto no ocurre pasamos al siguiente paso.
5. Construimos una nueva sustitución que vincule la variable con el término de D_k. Sea esta sustitución σ_{k+1}. Construimos ahora dos nuevos términos $E_{k+1} = \sigma_{k+1}(E_k)$ y $F_{k+1} = \sigma_{k+1}(F_k)$ y volvemos al paso 1.

Este algoritmo siempre termina para dos términos cualesquiera. Si los términos no eran unificables terminará indicándose así y si eran unificables devolverá un unificador de máxima generalidad y el término resultante unificado.

Algoritmo de Robinson

Ejemplo

1. Sean los términos p(a, X) y p(X , Y).

Algoritmo de Robinson

Ejemplo

1. Sean los términos p(a, X) y p(X , Y).

Sean E y F dos términos que queremos unificar. Consideramos inicialmente $\sigma_0 = \{\}$ una sustitución vacía, es decir, que no cambia ninguna variable. Dado que vamos a realizar un proceso iterativo, consideramos inicialmente $E_0 = \sigma_0(E)$ y $F_0 = \sigma_0(F)$. En cada iteración k del algoritmo se realizan los siguientes pasos:

1. Si $E_k = F_k$ entonces las cláusulas E y F son unificables y un unificador de máxima generalidad es $\sigma = \sigma_k \circ \cdots \circ \sigma_0$. Además, el término E_k es el término unificado. En este caso el proceso termina aquí.
2. Si $E_k = F_k$ entonces se busca el primer par de discordancia entre E_k y F_k. Sea éste D_k.
3. Si D_k contiene una variable y un término (pueden ser dos variables y una de ellas hace de término) pasamos al siguiente paso. En otro caso los términos no son unificables y terminamos el proceso.
4. Si la variable aparece en el término se produce un ***occur check*** por lo que E y F no unifican y terminamos. Si esto no ocurre pasamos al siguiente paso.
5. Construimos una nueva sustitución que vincule la variable con el término de D_k. Sea esta sustitución σ_{k+1}. Construimos ahora dos nuevos términos $E_{k+1} = \sigma_{k+1}(E_k)$ y $F_{k+1} = \sigma_{k+1}(F_k)$ y volvemos al paso 1.

LPO - **Método de Resolución**

- Lo que queremos demostrar es:

$$¿\Sigma \models \varphi?$$

- Equivalente a mostrar que:

$$\Sigma \cup \{\neg\varphi\} \text{ es contradictorio (i.e. } \Sigma \models \square \text{)}$$

Nota: decimos que □ es la cláusula vacía porque una **cláusula sin literales** no es satisfacible.

LPO - **Método de Resolución**

- Nos valdremos de la siguiente Regla de Resolución:

$$\frac{\begin{array}{c} p_1 + \ldots + p_j + \ldots + p_m \\ q_1 + \ldots + q_k + \ldots + q_n \end{array}}{(p_1 + \ldots + p_j + \ldots + p_m + q_1 + \ldots + q_k + \ldots + q_n)\boldsymbol{\theta}}$$

Donde p_j y q_k son uno la negación del otro (complementarios) y θ es su unificador

LPO - **Método de Resolución**

Suponga que quiere demostrar que φ es consecuencia lógica de Σ:

- Transforme Σ U {¬φ} a Forma Normal de Skolem.
- Usando la equivalencia $\forall x\, (A(x) * B(x)) \equiv \forall x\, A(x) * \forall x\, B(x)$
  - Transforme las fórmulas a un conjunto de cláusulas $C = \{c_1, ..., c_n\}$ (sin cuantificadores)
  - Mientras □ no pertenezca a C y existen c_j y c_k en C tales que la regla de resolución es aplicable:
    - Aplique la regla de resolución a c_j y c_k generando C'
    - Hacer C = C U {C'}

LPO - **Método de Resolución**

Ejercicio: Demostrar que

$$\models \exists x\, (P(x) \rightarrow \forall y\, P(y))$$

LPO - **Método de Resolución**

Ejercicio: Demostrar que

$$\models \exists x\, (P(x) \rightarrow \forall y\, P(y))$$

$$\Sigma \cup \{\neg\varphi\} = \{\ \} \cup \{\neg \exists x\, (P(x) \rightarrow \forall y\, P(y))\}$$

LPO - **Método de Resolución**

Ejercicio: Demostrar que

$$\models \exists x\,(P(x) \rightarrow \forall y\,P(y))$$

$$\Sigma \cup \{\neg\varphi\} = \{\ \} \cup \{\neg \exists x\,(P(x) \rightarrow \forall y\,P(y))\}$$

Transformamos las fórmulas a un conjunto de cláusulas

$$\neg \exists x\,(P(x) \rightarrow \forall y\,P(y))$$

LPO - **Método de Resolución**

Ejercicio: Demostrar que

$$\models \exists x\,(P(x) \rightarrow \forall y\, P(y))$$

$$\Sigma \cup \{\neg\varphi\} = \{\,\} \cup \{\neg \exists x\,(P(x) \rightarrow \forall y\, P(y))\}$$

Transformamos las fórmulas a un conjunto de cláusulas

$$\neg \exists x\,(P(x) \rightarrow \forall y\, P(y))$$

$$\equiv \neg \exists x\,(\neg P(x) \lor \forall y\, P(y))$$ [eliminamos →]

LPO - **Método de Resolución**

Ejercicio: Demostrar que

$$\models \exists x\,(P(x) \rightarrow \forall y\,P(y))$$

$$\Sigma \cup \{\neg\varphi\} = \{\,\} \cup \{\neg \exists x\,(P(x) \rightarrow \forall y\,P(y))\}$$

Transformamos las fórmulas a un conjunto de cláusulas

$$\neg \exists x\,(P(x) \rightarrow \forall y\,P(y))$$

$$\equiv \neg \exists x\,(\neg P(x) \vee \forall y\,P(y))$$ [eliminamos →]

$$\equiv \forall x\,\neg(\neg P(x) \vee \forall y\,P(y))$$ [ingresamos ¬]

LPO - **Método de Resolución**

Ejercicio: Demostrar que

$$\models \exists x\,(P(x) \rightarrow \forall y\, P(y))$$

$$\Sigma \cup \{\neg\varphi\} = \{\,\} \cup \{\neg \exists x\,(P(x) \rightarrow \forall y\, P(y))\}$$

Transformamos las fórmulas a un conjunto de cláusulas

$$\neg \exists x\,(P(x) \rightarrow \forall y\, P(y))$$

$$\equiv \neg \exists x\,(\neg P(x) \lor \forall y\, P(y)) \qquad \text{[eliminamos } \rightarrow]$$

$$\equiv \forall x\, \neg(\neg P(x) \lor \forall y\, P(y)) \qquad \text{[ingresamos } \neg]$$

$$\equiv \forall x\, (P(x) \land \neg \forall y\, P(y)) \qquad \text{[ingresamos } \neg]$$

LPO - **Método de Resolución**

Ejercicio: Demostrar que

$$\models \exists x\,(P(x) \rightarrow \forall y\, P(y))$$

$$\Sigma \cup \{\neg\varphi\} = \{\ \} \cup \{\neg \exists x\,(P(x) \rightarrow \forall y\, P(y))\}$$

Transformamos las fórmulas a un conjunto de cláusulas

$$\neg \exists x\,(P(x) \rightarrow \forall y\, P(y))$$

$$\equiv \neg \exists x\,(\neg P(x) \lor \forall y\, P(y)) \qquad \text{[eliminamos} \rightarrow\text{]}$$

$$\equiv \forall x\, \neg(\neg P(x) \lor \forall y\, P(y)) \qquad \text{[ingresamos } \neg\text{]}$$

$$\equiv \forall x\, (P(x) \land \neg \forall y\, P(y)) \qquad \text{[ingresamos } \neg\text{]}$$

$$\equiv \forall x\, (P(x) \land \exists y\, \neg P(y)) \qquad \text{[ingresamos } \neg\text{]}$$

LPO - **Método de Resolución**

Ejercicio: Demostrar que

$$|= \exists x\,(P(x) \rightarrow \forall y\, P(y))$$

$$\Sigma \cup \{\neg\varphi\} = \{\,\} \cup \{\neg \exists x\,(P(x) \rightarrow \forall y\, P(y))\}$$

Transformamos las fórmulas a un conjunto de cláusulas

$$
\begin{aligned}
& \neg \exists x\,(P(x) \rightarrow \forall y\, P(y)) & \\
\equiv\; & \neg \exists x\,(\neg P(x) \lor \forall y\, P(y)) & \text{[eliminamos } \rightarrow\text{]} \\
\equiv\; & \forall x\, \neg(\neg P(x) \lor \forall y\, P(y)) & \text{[ingresamos } \neg\text{]} \\
\equiv\; & \forall x\, (P(x) \land \neg \forall y\, P(y)) & \text{[ingresamos } \neg\text{]} \\
\equiv\; & \forall x\, (P(x) \land \exists y\, \neg P(y)) & \text{[ingresamos } \neg\text{]} \\
\equiv\; & \forall x\, \exists y\, (P(x) \land \neg P(y)) & \text{[exteriorizar } \exists y\text{]}
\end{aligned}
$$

LPO - **Método de Resolución**

Ejercicio: Demostrar que

$$\models \exists x\,(P(x) \rightarrow \forall y\,P(y))$$

$$\Sigma \cup \{\neg\varphi\} = \{\ \} \cup \{\neg \exists x\,(P(x) \rightarrow \forall y\,P(y))\}$$

Transformamos las fórmulas a un conjunto de cláusulas

$$\neg \exists x\,(P(x) \rightarrow \forall y\,P(y))$$

$$\equiv \neg \exists x\,(\neg P(x) \lor \forall y\,P(y)) \qquad \text{[eliminamos } \rightarrow]$$

$$\equiv \forall x\,\neg(\neg P(x) \lor \forall y\,P(y)) \qquad \text{[ingresamos } \neg]$$

$$\equiv \forall x\,(P(x) \land \neg \forall y\,P(y)) \qquad \text{[ingresamos } \neg]$$

$$\equiv \forall x\,(P(x) \land \exists y\,\neg P(y)) \qquad \text{[ingresamos } \neg]$$

$$\equiv \forall x\,\exists y\,(P(x) \land \neg P(y)) \qquad \text{[exteriorizar } \exists y]$$

$$\equiv \forall x\,(P(x) \land \neg P(f(x))) \qquad \text{[skolemizar]}$$

LPO - **Método de Resolución**

$$C = \{P(x), \neg P(f(x))\}$$

LPO - **Método de Resolución**

$$C = \{P(x), \neg P(f(x))\}$$

es conveniente renombrar las variables

LPO - **Método de Resolución**

$$C = \{P(x), \neg P(f(x))\}$$

es conveniente renombrar las variables… **para encontrar unificaciones**

LPO - **Método de Resolución**

$$C = \{P(x), \neg P(f(x))\}$$

es conveniente renombrar las variables... **para encontrar unificaciones**

$$C = \{P(x), \neg P(f(\mathbf{y}))\}$$

LPO - **Método de Resolución**

$$C = \{P(x), \neg P(f(x))\}$$

es conveniente renombrar las variables... **para encontrar unificaciones**

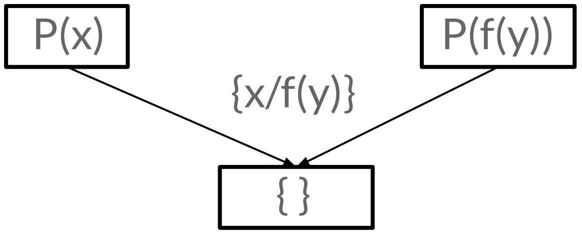
$$C = \{P(x), \neg P(f(\mathbf{y}))\}$$

una aplicación de la regla de resolución permite obtener:

$$\frac{\begin{array}{c} P(x)\ \mathbf{\{x/f(y)\}} \\ \neg P(f(y)) \end{array}}{\square}$$

LPO - **Método de Resolución**

Otra forma de representar la aplicación de la regla de resolución es mostrarlo como un **Grafo Acíclico Dirigido** (GAD)

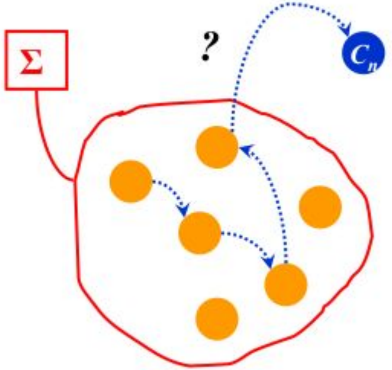


LPO - **Un sistema completo**

El sistema basado en resolución sólo nos permite usar cláusulas.

Sólo podemos usar resolución para demostrar que ¬C es inconsistente.

LPO - **Método de Resolución**



- Dados:
  - un conjunto de cláusulas Σ
  - una cláusula C

Una demostración por resolución de C desde Σ es una secuencia de cláusulas $C_1, C_2, ..., C_n$ tal que:

- Para cada $i \leq n$:
  - C_i pertenece a Σ o
  - C_i es una tautología o
  - C_i es obtenido por aplicación de regla de resolución a partir de C_j y C_k
- $C_n = C$

$$\Sigma \vdash_{Res.} C$$

LPO - **Método de Resolución**

- **Teorema**: (Completitud de Resolución) Dado un conjunto de cláusulas $\Sigma \cup \{C\}$

$$\text{si } \Sigma \models C \text{ entonces } \Sigma \vdash_{Res.} C$$

- **Teorema**: (Correctitud de Resolución) Si C se puede deducir desde Σ usando el conjunto de reglas, entonces C es consecuencia lógica de Σ. En símbolos:

$$\text{si } \Sigma \vdash_{Res.} C \text{ entonces } \Sigma \models C$$

LPO - **Método de Resolución**

- **Teorema**: (Completitud de Resolución) Dado un conjunto de cláusulas $\Sigma \cup \{C\}$

$$\text{si } \Sigma \models C \text{ entonces } \Sigma \vdash_{\text{Res.}} C$$

- **Teorema**: (Correctitud de Resolución) Si C se puede deducir desde Σ usando el conjunto de reglas, entonces C es consecuencia lógica de Σ. En símbolos:

$$\text{si } \Sigma \vdash_{\text{Res.}} C \text{ entonces } \Sigma \models C$$

Finalmente:

$$\boldsymbol{\Sigma \vdash_{\text{Res.}} C \iff \Sigma \models C}$$

LPO - **Un sistema completo**

El sistema basado en resolución sólo nos permite usar cláusulas.

¿Cómo generar un sistema completo para cualquier tipo de fórmula?

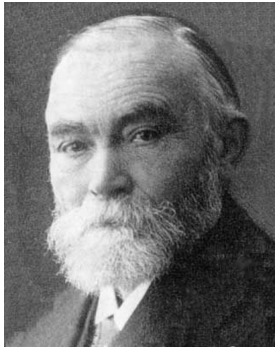
demostración*...

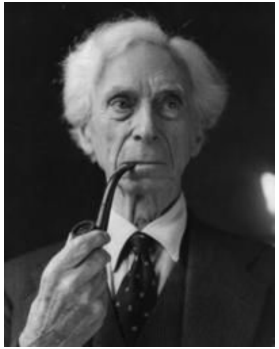
- Múltiples tentativas a comienzos del siglo XX de fundamentar la matemática sobre sólidas bases lógicas.

demostración*...

- Múltiples tentativas a comienzos del siglo XX de fundamentar la matemática sobre sólidas bases lógicas.

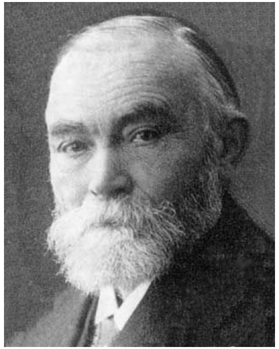
    - Frege y Russell intentan reducir las matemáticas a la lógica y la teoría de conjuntos.



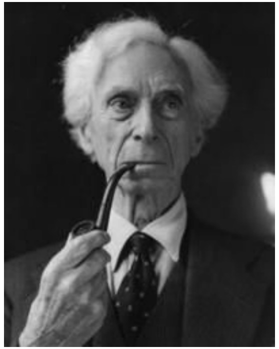


demostración*...

- Múltiples tentativas a comienzos del siglo XX de fundamentar la matemática sobre sólidas bases lógicas.
  - Frege y Russell intentan reducir las matemáticas a la lógica y la teoría de conjuntos.



PARADOJA



demostración*...

En un lejano poblado de un antiguo emirato había un barbero llamado As-Samet *diestro en afeitar cabezas y barbas, maestro en escamondar pies y en poner sanguijuelas*. Un día el emir se dio cuenta de la falta de barberos en el emirato, y ordenó que los barberos sólo afeitaran a aquellas personas que no pudieran afeitarse. Y así mismo impuso la norma de que todo el mundo se afeitase, (no se sabe si por higiene, por estética, o por demostrar que podía imponer su santa voluntad y mostrar así su poder). Cierto día el emir llamó a As-Samet para que lo afeitara y él le contó sus angustias:

—En mi pueblo soy el único barbero. No puedo afeitar al barbero de mi pueblo, ¡que soy yo!, ya que si lo hago, entonces puedo afeitarme por mí mismo, por lo tanto ¡no debería afeitarme! pues desobedecería vuestra orden. Pero, si por el contrario no me afeito, entonces algún barbero debería afeitarme, ¡pero como yo soy el único barbero de allí!, no puedo hacerlo y también así desobedecería a vos mi señor, oh emir de los creyentes, ¡que Allah os tenga en su gloria!

El emir pensó que sus pensamientos eran tan profundos, que lo premió con la mano de la más virtuosa de sus hijas. Así, el barbero As-Samet vivió para siempre feliz y barbón.[2]

demostración*...

- Múltiples tentativas a comienzos del siglo XX de fundamentar la matemática sobre sólidas bases lógicas.
  - Frege y Russell intentan reducir las matemáticas a la lógica y la teoría de conjuntos.

demostración*...

- Múltiples tentativas a comienzos del siglo XX de fundamentar la matemática sobre sólidas bases lógicas.
  - Frege y Russell intentan reducir las matemáticas a la lógica y la teoría de conjuntos.
- Otro intento fue el de David Hilbert (1862-1943), iniciador de la teoría de la demostración o metamatemática.
  - Buscaba encontrar un modo de demostrar la **consistencia** de cualquier lista de axiomas.

demostración*...

- Múltiples tentativas a comienzos del siglo XX de fundamentar la matemática sobre sólidas bases lógicas.
  - Frege y Russell intentan reducir las matemáticas a la lógica y la teoría de conjuntos.
- Otro intento fue el de David Hilbert (1862-1943), iniciador de la teoría de la demostración o metamatemática.
  - Buscaba encontrar un modo de demostrar la **consistencia** de cualquier lista de axiomas.
- Hilbert tenía una concepción de las matemáticas que denominaba formalismo.
  - Las cosas de las que hablan las matemáticas no son más que símbolos.
  - Los símbolos carecen de significado por sí mismos: Lo sabemos todo sobre ellos cuando comenzamos a manipularlos.
  - Estableció reglas recursivas para explicar sus posibles interacciones.

demostración*...

El sistema basado en resolución sólo nos permite usar cláusulas.

demostración*...

El sistema basado en resolución sólo nos permite usar cláusulas.

¿Cómo generar un sistema completo para cualquier tipo de fórmula?

demostración*...

El sistema basado en resolución sólo nos permite usar cláusulas.

¿Cómo generar un sistema completo para cualquier tipo de fórmula?

Vamos a mostrar un sistema completo para

$$\{\neg, \rightarrow\}$$

demostración*...

Resolución da una idea de qué es lo que se necesita para tener un sistema completo:

- Un conjunto de reglas para inferir tautologías.
- Un conjunto de reglas de inferencia.

David Hilbert diseñó un sistema completo con estos componentes.

demostración*...

Algunas fórmulas que **son** universalmente válidas en LPO:

$$\forall x P(x) \rightarrow P(C)$$

$$P(y) \rightarrow \exists x P(x)$$

$$\forall x P(x) \rightarrow \exists y P(y)$$

$$\neg \forall x P(x) \rightarrow \exists x \neg P(x)$$

demostración*...

Sin embargo, éstas fórmulas **no son** universalmente válidas:

$$\exists x P(x) \rightarrow \forall x P(x)$$

$$\forall x (P(x) \rightarrow P(C))$$

$$\exists x \neg P(x) \rightarrow \neg \exists y P(y)$$

demostración*...

Cualquier tautología de lógica proposicional puede ser llevada a una fórmula universalmente válida en LPO.

$$\models P(x) \rightarrow (Q(x) \rightarrow P(x))$$

demostración*...

El sistema de deducción de Hilbert consta de los siguientes elementos:

- Esquemas para generar fórmulas válidas (los cuales también son fórmulas válidas).
- Axiomas para la igualdad.
- Reglas de Inferencia.

demostración*...

Esquemas para generar fórmulas válidas

(a) $(\alpha \rightarrow (\beta \rightarrow \alpha))$

(b) $(\alpha \rightarrow (\beta \rightarrow \gamma)) \rightarrow ((\alpha \rightarrow \beta) \rightarrow (\alpha \rightarrow \gamma))$

(c) $(\neg\alpha \rightarrow \neg\beta) \rightarrow ((\neg\alpha \rightarrow \beta) \rightarrow \alpha)$

(d) $(\neg\beta \rightarrow \neg\alpha) \rightarrow (\alpha \rightarrow \beta)$

(e) $(\forall x\varphi(x)) \rightarrow \varphi(t)$ donde t es un término cualquiera.

(f) $\varphi(t) \rightarrow (\exists x\varphi(x))$ donde t es un término cualquiera.

(g) $(\exists x\varphi) \rightarrow (\neg\forall x\neg\varphi)$

demostración*...

Axiomas para la igualdad

(a) $\forall x(x = x)$

(b) $\forall x \forall y(x = y \rightarrow y = x)$

(c) $\forall x \forall y \forall z((x = y \cdot y = z) \rightarrow x = z)$

(d) Para todo predicado *m*-ario P:

$$\forall \overrightarrow{\mathbf{x}} \forall \overrightarrow{\mathbf{y}} ((P(\overrightarrow{\mathbf{x}}) \cdot \overrightarrow{\mathbf{x}} = \overrightarrow{\mathbf{y}}) \rightarrow P(\overrightarrow{\mathbf{y}}))$$

(e) Para toda función *n*-aria f :

$$\forall \overrightarrow{\mathbf{x}} \forall \overrightarrow{\mathbf{y}} (\overrightarrow{\mathbf{x}} = \overrightarrow{\mathbf{y}} \rightarrow f(\overrightarrow{\mathbf{x}}) = f(\overrightarrow{\mathbf{y}}))$$

LPO - **Sistema Deductivo de Hilbert**

Reglas de Inferencia

(a) Modus Ponens:

$$\frac{\begin{array}{c}\alpha \rightarrow \beta \\ \alpha\end{array}}{\beta}$$

(b) Generalización: si *y* no aparece libre en φ

$$\frac{\varphi \rightarrow \psi(y)}{\varphi \rightarrow \forall y \psi(x)}$$

demostración...

Dado un conjunto de fórmulas $\Sigma \cup \{\varphi\}$, una deducción formal de φ desde Σ es una secuencia de fórmulas $\varphi_1, \varphi_2, \ldots \varphi_n$ tal que:

para cada $i \leq n$:

- $\varphi_i \in \Sigma$ ó
- $\varphi_i \in$ es un axioma lógico (de los ya presentados) ó
- $\varphi_i \in$ es obtenido a través de Modus Ponens desde φ_j y φ_k ó
- $\varphi_i \in$ es obtenido a través de la Regla de Generalización desde φ_j y φ_k
- $\varphi_n = \varphi$

demostración...

Dado un conjunto de fórmulas $\Sigma \cup \{\varphi\}$, una deducción formal de φ desde Σ es una secuencia de fórmulas $\varphi_1, \varphi_2, \ldots \varphi_n$ tal que:

para cada $i \leq n$:

- $\varphi_i \in \Sigma$ ó
- $\varphi_i \in$ es un axioma lógico (de los ya presentados) ó
- $\varphi_i \in$ es obtenido a través de Modus Ponens desde φ_j y φ_k ó
- $\varphi_i \in$ es obtenido a través de la Regla de Generalización desde φ_j y φ_k
- $\varphi_n = \varphi$

➡ Cuando φ se obtiene de una demostración formal a partir de Σ, decimos que $\boldsymbol{\Sigma \vdash_H \varphi}$

demostración...

- Quería que la matemática fuese formulada sobre bases sólidas y completamente lógicas.

demostración...

- Quería que la matemática fuese formulada sobre bases sólidas y completamente lógicas.
- En 1920 propuso un proyecto conocido como *programa de Hilbert*.... el reto de demostrar que los axiomas de la aritmética son consistentes, es decir, no conducen a ninguna contradicción.

demostración...

- Quería que la matemática fuese formulada sobre bases sólidas y completamente lógicas.
- En 1920 propuso un proyecto conocido como *programa de Hilbert*.... el reto de demostrar que los axiomas de la aritmética son consistentes, es decir, no conducen a ninguna contradicción.
- El programa comprendía demostrar que:
    - Todo conocimiento matemático se deriva de un conjunto finito de axiomas cuidadosamente seleccionados, y
    - Puede demostrarse que tal sistema axiomático es consistente, es decir, no conduce a contradicciones cuando se derivan conclusiones de ellos.

demostración...

Camino al concepto de algoritmo....

- Galileo y las leyes naturales descritas mediante la matemática.

demostración...

Camino al concepto de algoritmo….

- Galileo y las leyes naturales descritas mediante la matemática.

- Aparición del supuesto que "todo problema enunciado en términos matemáticos, podría resolverse siguiendo una serie de pasos estipulados, ya fuese en forma geométrica o de tipo más algebraico".

demostración...

Camino al concepto de algoritmo….

Calculadora Pascalina 1642

- Galileo y las leyes naturales descritas mediante la matemática.
- Aparición del supuesto que "todo problema enunciado en términos matemáticos, podría resolverse siguiendo una serie de pasos estipulados, ya fuese en forma geométrica o de tipo más algebraico".
- Leibniz en el siglo XVII y su máquina mecánica de cálculo.

demostración...

- Para Hilbert era crucial que todo problema tuviese una solución exacta, cuya única dificultad rayara simplemente en encontrarla, pero sobre cuya existencia no hubiese duda.

- Este problema fue formalizado primero parcialmente, a través de su décimo problema de la Conferencia de París, y ya en forma más elaborada en el VIII Congreso Internacional de Matemática, celebrado en Bolonia en 1928.

demostración...

- Hilbert planteó la búsqueda de un procedimiento algorítmico general válido para resolver todas las posibles cuestiones matemáticas.
- Su planteamiento buscaba obtener la respuesta a tres importantes preguntas:
  - **¿Son las matemáticas completas?** (cualquier proposición puede ser probada o rechazada)
  - **¿Son las matemáticas consistentes?** (no se producen contradicciones)
  - **¿Son las matemáticas decidibles?** (cualquier proposición se puede demostrar tras una secuencia finita de pasos)

demostración...

- Hilbert planteó la búsqueda de un procedimiento algorítmico general válido para resolver todas las posibles cuestiones matemáticas.
- Su planteamiento buscaba obtener la respuesta a tres importantes preguntas:
    - **¿Son las matemáticas completas?** (cualquier proposición puede ser probada o rechazada)
    - **¿Son las matemáticas consistentes?** (no se producen contradicciones)
    - **¿Son las matemáticas decidibles?** (cualquier proposición se puede demostrar tras una secuencia finita de pasos)

completitud...

- Los planteamientos de Hilbert atrajeron una multitud de investigadores, entre ellos Kurt Gödel (1906 – 1978).

El problema de la completitud consiste en demostrar que **un sistema axiomático**, del tipo de los que Hilbert propuso, es capaz de demostrar o derivar toda proposición verdadera dentro del sistema.

- En 1929 demostró la completitud del cálculo de predicados de primer orden o lógica de primer orden.

Ninguna proposición verdadera podía escapar del poder demostrativo de este tipo de lógica.

demostración...

- Hilbert planteó la búsqueda de un procedimiento algorítmico general válido para resolver todas las posibles cuestiones matemáticas.
- Su planteamiento buscaba obtener la respuesta a tres importantes preguntas:
  - **¿Son las matemáticas completas?** (cualquier proposición puede ser probada o rechazada)
  - **¿Son las matemáticas consistentes?** (no se producen contradicciones)
  - **¿Son las matemáticas decidibles?** (cualquier proposición se puede demostrar tras una secuencia finita de pasos)

Incompletitud

- La **incompletitud** implicaba la falta de axiomas, pues se suponía que una proposición verdadera no demostrable podría serlo agregando nuevos axiomas.
- En 1931 demostró que cualquier sistema axiomático computable que sea capaz de describir la aritmética de los números naturales (por ej. los Axiomas de Peano), está sujeto a las siguientes condiciones:
    - Si el sistema es consistente entonces no puede ser completo, y
    - La consistencia de los axiomas no puede demostrarse desde dentro del sistema.

Incompletitud

- La **incompletitud** implicaba la falta de axiomas, pues se suponía que una proposición verdadera no demostrable podría serlo agregando nuevos axiomas.
- En 1931 demostró que cualquier sistema axiomático **computable** que sea capaz de describir la aritmética de los números naturales (por ej. los Axiomas de Peano), está sujeto a las siguientes condiciones:
  - Si el sistema es consistente entonces no puede ser completo, y
  - La consistencia de los axiomas no puede demostrarse desde dentro del sistema.

Hilbert, we have a problem!

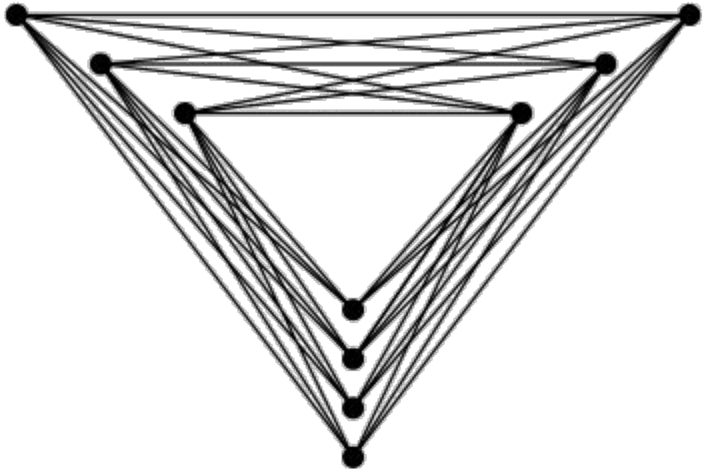
demostración...

- Hilbert planteó la búsqueda de un procedimiento algorítmico general válido para resolver todas las posibles cuestiones matemáticas.
- Su planteamiento buscaba obtener la respuesta a tres importantes preguntas:
  - ¿Son las matemáticas completas?, es decir cualquier proposición puede ser probada o rechazada
  - ¿Son las matemáticas consistentes?, es decir no es posible demostrar algo falso
  - **¿Son las matemáticas decidibles?, es decir cualquier proposición se puede demostrar como cierta o falsa tras una secuencia finita de pasos**

Departamento de Ingeniería Informática
Universidad de Santiago de Chile

Teoría de la Computación

Primer semestre 2024



Santiago, 2024